Some contributions to nonlinear adaptive control of PKMs : from design to real-time experiments

Moussab Bennehar

Collège Doctoral du Languedoc-Roussillon

THÈSE

Pour obtenir le grade de Docteur

Délivré par l'**Université de Montpellier**

Préparée au sein de l'école doctorale **I2S***
Et de l'unité de recherche **UMR 5506**

Spécialité : **Systèmes Automatiques etMicroélectroniques**

Présentée par **Moussab Bennehar**

Some Contributions to Nonlinear Adaptive Control of PKMs : From Design to Real-Time Experiments

Soutenue le 17/12/2015 devant le jury composé de :

Antoine Ferreira	Professeur	INSA Centre Val de Loire	Rapporteur
Nacer Kouider M'sirdi	Professeur	Université d'Aix Marseille	Rapporteur
Mohamed Bouri	Maître de conf.	École polytech. fédérale de Lausanne	Examinateur
Sébastien Krut	CR CNRS	LIRMM	Examinateur
François Pierrot	DR CNRS	LIRMM	Direct. de thèse
Ahmed Chemori	CR CNRS	LIRMM	Encadrant

Université Montpellier

* **I2S :** École doctorale Information Structures Systèmes

Contents

List of Figures

List of Tables

Acknowledgements

Firstly, I would like to express my sincere gratitude to my supervisor, Ahmed Chemori, for his continuous support of my PhD study during the past three years, for his patience, motivation, and immense knowledge. His advice on both research as well as on my career have been priceless. I could not have imagined having a better supervisor and mentor for my PhD study.

I would also like to thank my committee members, Antoine Ferreira, Nacer Kouider Msirdi, Mohamed Bouri and Sébastien Krut for their presence and for their insightful comments and encouragement. Thank you for letting my defense be an enjoyable moment.

I would also like to thank François Pierrot for accepting to be my thesis director, for giving me the opportunity to join LIRMM and accomplish this thesis.

I would especially like to thank all my colleagues at LIRMM. A special thanks goes to my dear PhD colleagues Gamal and Mohamed. Thank you for your support and encouragement, for the sleepless nights we were working together before deadlines and for all the fun we have had in the last three years.

A special thanks to my family. Words cannot express how grateful I am to my parents, my brothers and my sister.

At the end I would like to express my appreciation to my beloved wife Sarra who spent sleepless nights with me and was always my support in the moments when there was no one to answer my queries.

Introduction

Parallel Kinematic Manipulators (PKMs) have gained an increasing popularity in the last few decades. This interest has been stimulated by the significant advantages of PKMs over their traditional serial counterparts regarding some specific industrial tasks requiring high accelerations and high accuracy. In fact, a duality exists between PKMs and serial manipulators such that PKMs excel in areas where serial manipulators do not perform well, and vice versa. Even though conventional serial manipulators possess better dexterity and larger workspace than PKMs, their open kinematic chain structure makes them subject to bending when carrying high loads and vibrations when operating at high accelerations, resulting in a lack of precision and stiffness. Consequently, PKMs are promoted as a better alternative in robotic tasks where high payload, precise positioning and high accelerations are needed. Such applications may include, among others, pick-and-place tasks, high-speed machining, precision surgery, etc.

With the increasing interest in PKMs both in industry and academia, the potential that these mechanisms hold has been progressively unclouded. However, to fully exploit their potential and to get the most of their capabilities, a long path is still to be covered. In addition to mechanical design, calibration and optimization of the structure, efficient control development plays an essential role in improving the performance of PKMs. Indeed, PKMs are known for their highly nonlinear dynamics which increases considerably when operating at high accelerations leading to mechanical vibrations and lack of precision. Moreover, uncertainties are abundant in such systems due to model simplifications, the wear of the components of the robot, variations of the environment, etc. Furthermore, their coupled actuation and the actuation redundancy in some platforms give rise to complex and challenging control issues. Hence, in order to take full advantage of their dynamic capabilities and achieve the desired tracking accuracy performance, advanced and sophisticated con-

trol schemes that tie with the complexity, the uncertainty and the variable nature of their dynamics have to be developed.

Considering the above arguments and having established the necessity of designing advanced control algorithms to yield the desired tracking performance, one question arises: "What are the most suitable control techniques that are capable of handling such complex and uncertain dynamics while yielding satisfactory tracking performance?". Certainly, control schemes that take advantage from the modeled dynamics of the robot in the control loop have demonstrated their superiority throughout the years when it comes to control PKMs operating at high accelerations. Moreover, adaptive techniques can be adopted in order to account for uncertainties and disturbances that may be encountered by the robot in a typical industrial task. Therefore, we focus our attention in this dissertation on adaptive control schemes that considers the modeled dynamics of the robot with the aim of improving its tracking capabilities while operating at high accelerations.

The main goal of this thesis lies in the proposal of advanced adaptive control schemes to be implemented on PKMs operating at high accelerations. We focus our attention on improving the tracking capabilities of PKMs by accounting for the uncertainties and disturbances that may be encountered in a real industrial application. Additionally, one of the main objectives of this thesis is to surpass theoretical developments and to experimentally validate the proposed control strategies through real-time experiments. Hence, the proposed control schemes in this thesis are substantiated with real-time experiments on different types of PKMs prototypes.

Publications of the author

Accepted for publication in international journals

[JP1] M. Bennehar, A. Chemori, S. Krut, and F. Pierrot. Control of Redundantly Actuated PKMs for Closed-Shape Trajectories Tracking with Real-Time Experiments. Transactions on Systems, Signals and Devices, 2015. To appear.

[JP2] M. Bennehar, A. Chemori, F. Pierrot and V. Creuze. Extended Model-Based Feedforward Compensation in $\mathcal{L}_1$ Adaptive Control for Mechanical Manipulators: Design and Experiments. Frontiers in Robotics and AI.

Published in international conferences proceedings

[IC1] M. Bennehar, A. Chemori, and F. Pierrot. $\mathcal{L}_1$ Adaptive Control of Parallel Kinematic Manipulators: Design and Real-Time Experiments. In IEEE International Conference on Robotics and Automation (ICRA15), pages 1587-1592, Seattle, Washington, USA, May 2015.

[IC2] M. Bennehar, A. Chemori, and F. Pierrot. A Novel RISE-Based Adaptive Feedforward Controller for Redundantly Actuated Parallel Manipulators. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS14), pages 2389-2394, Chicago, Illinois, USA, September 2014.

[IC3] M. Bennehar, A. Chemori, and F. Pierrot. A New Extension of Direct Compensation Adaptive Control and Its Real-Time Application to Redundantly Actuated PKMs. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS14), pages 1670-1675, Chicago, Illinois, USA, September 2014.

[IC4] M. Bennehar, A. Chemori, S. Krut, and F. Pierrot. Continuous Closed-Form Trajectories Generation and Control of Redundantly Actuated Parallel Kinematic Manipulators. In Multi-Conference on Systems, Signals and Devices (SSD14), Barcelona, Spain, 2014.

Submitted to international journals

[JS1] M. Bennehar, A. Chemori, M. Bouri, L.F Jenni and F. Pierrot. A New Adaptive RISE-Based Control for Parallel Robots: Design, Stability Analysis and Experiments. International Journal of Control.

[JS2] M. Bennehar, A. Chemori, and F. Pierrot. A New Revised Desired Compensation Adaptive Control for Enhanced Tracking: Application to RA-PKMs. Advanced Robotics.

[JS3] M. Bennehar, A. Chemori, and F. Pierrot. Feedforward Augmented $\mathcal{L}_1$ Adaptive Controller for Parallel Kinematic Manipulators with Improved Tracking. IEEE Robotics and Automation Letters (RA-L).

Organization of the thesis

The contents of this dissertation are organized into three parts as follows

Part 1 includes two chapters and is devoted to the context of this thesis, the control problem formulation as well as the state of the art on control of PKMs.

Chapter 1 presents the context and the problem formulation of the current thesis. It includes a brief history of PKMs as well as some of the most successful parallel robots that have been commercialized throughout the years. It concludes with the challenging problems that could be encountered when dealing with control of PKMs.

Chapter 2 is dedicated to a state of the art on main control strategies that have been proposed in the literature for PKMs. The different control strategies are divided into several classes according to their complexity and adaptive capabilities.

Part 2 is devoted to the main contributions of this thesis; namely, the proposed control schemes, and is composed of two chapters.

Chapter 3 presents the first two proposed controllers within this thesis. These controllers are capable of adaptively dealing with uncertainties in the dynamics of the manipulator. Furthermore, their extension to PKMs with actuation redundancy is provided at the end of the chapter.

Chapter 4 addresses a new class of adaptive controllers called $\mathcal{L}_1$ adaptive control. This strategy, known for its fast adaptation and robustness qualities, has never been experimentally validated on PKMs. Then, two novel extensions for $\mathcal{L}_1$ adaptive control are presented in order to yield improved tracking performance.

Part 3 addresses the experimental validation of the proposed control schemes presented in **Part 2** and is divided into three chapters.

Chapter 5 is dedicated to the presentation of experimental platforms used during this thesis for experimental validation of the proposed controllers. These platforms are classified according to their actuation type (redundant and non-redundant).

Chapter 6 is dedicated to the real-time validation of the first developed controllers discussed in **Chapter 3**. Two scenarios are considered for each control scheme to demonstrate its robustness towards payload changes.

Chapter 7 addresses the real-time validation of $\mathcal{L}_1$ adaptive control as well as the two proposed extensions in **Chapter 4**. These control schemes are validated on two PKMs prototypes developed at LIRMM. The first one is non-redundant while the second one is redundant.

Part I

Context, problem formulation and state of the art

CHAPTER

1 Context and problem formulation

Contents

1.1 Introduction

This chapter is devoted to some historical aspects and generalities on PKMs as well as the context and the problem formulation of the thesis. First, let us start with a basic definition of a PKM. According to Merlet [Merlet, 2006], a generalized PKM can be defined as follows:

> "A generalized PKM is a closed-loop kinematic chain mechanism whose end-effector is linked to the base by several independent kinematic chains".

It is often mentioned that PKMs are attracting an increasing interest in robotics community. Indeed, PKMs exhibit several advantages compared to their serial counterparts. Such advantages however come accompanied with some limitations and challenges. Indeed, PKMs are known by their complex mechanical structure leading to challenging issues in terms of kinematic and dynamic modeling. From a control point of view, PKMs

are known to be multivariable, coupled and highly nonlinear systems. In contrast with linear ones, nonlinear systems are challenging require advanced control skills. In addition to their inherent highly nonlinear nature, the dynamics of PKMs is often uncertain and/or time varying. Because of these issues, and many others, control of PKMs is usually described in research as a challenging task.

This chapter starts with some history about PKMs, their origins and some of their typical industrial applications. Then, the context and the control problem formulation of this thesis are described.

1.2 Overview of parallel manipulators

Most researchers mention that PKMs are a recent technology proposed as a solution for the drawbacks of serial manipulators [Angeles, 2002]. They often refer to the hexapod designed by Gough in the late 1950s as the first PKM ever designed. However, their story dates back to many years before the invention of the Gough-platform. In what follows, a brief history about parallel manipulators as well as their evolution throughout the years is provided.

1.2.1 Origin of parallel manipulators

According to [Bonev, 2014], the first ever patented parallel manipulator dates back to 1928 and was based on a spherical structure. The patent which was filled by James E. Gwinnett consisted of an amusement device aimed to be used as a dynamic cinema (cf. Figure 1.1). Unfortunately, his invention was never built because it was too complex for the industry at that time. Ten years later, Willard L.V. Pollard designed a five **D**egree-**O**f-**F**reedom (5-DOF) parallel manipulator intended to be used as a spray painting machine. His invention, depicted in Figure 1.2, has later been patented by his son Willard L.V. Pollard Jr in 1942. While other spray painting robots have been commercialized, the one of Pollard has never been built.

The commercialization of PKMs indeed started after the development of the Gough platform (also referred to as the Stewart or Gough-Stewart platform) in 1947 [Gough and Whitehall, 1962]. His prototype, mainly used for tyre-testing, has been extensively studied in the literature and often referred to as the first parallel manipulator ever built.

Many years later, in 1965, Stewart introduced his 6-DOF parallel manipulator intended to be used as a flight simulator [Stewart, 1966]. His design was motivated by the burst of the aeronautic industry and, consequently, the increasing cost of the pilots' training, in addition to the need for testing new aircraft.

For almost two decades, parallel manipulators have not attracted as much attention as they did in the early 80s. Indeed, hundreds of parallel manipulators designs have been proposed in almost every industrial application. However, the design that has rev-

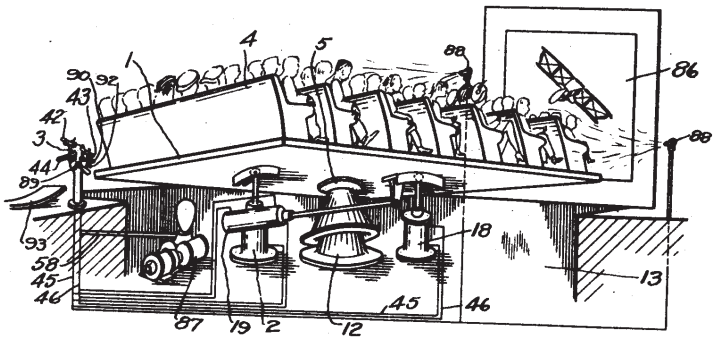


Figure 1.1 – Amusement device based on a spherical PKM proposed by James E. Gwinnett (US Patent 1,789,680) [Gwinnett, 1931]

olutionize parallel manipulators industry and attracted tremendous interest is the Delta robot [Clavel, 1985], designed by Raymond Clavel, a professor at the École Polytechnique Fédérale de Lausanne (EPFL). His innovative idea of using lightweight parallelograms for building high-speed PKMs (cf. Figure. 1.5) has attracted the attention of several companies. The Delta robot, typically used for high-speed pick-and-place applications of light objects, has been sold more than a thousand times, which makes it the most successful PKM ever built. The use of the Delta robots has expanded over the years covering multiple applications such as drilling, food packaging, watchmaking, 3D printing, etc. The idea of using parallelograms has inspired many new designs with similar features.

With the growing interest in PKMs, plenty of novel designs and prototypes of parallel manipulators have been developed. Even though they often share a lot of common similarities, each prototype usually has its own characteristics, advantages, drawbacks and targeted applications.

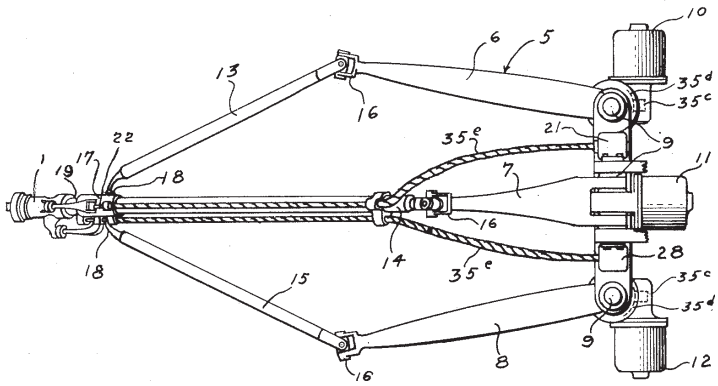


Figure 1.2 – Spray painting 5-DOF PKM proposed by Willard L.V. Willard Jr (US Patent 2,286,571) [Pollard, 1942]

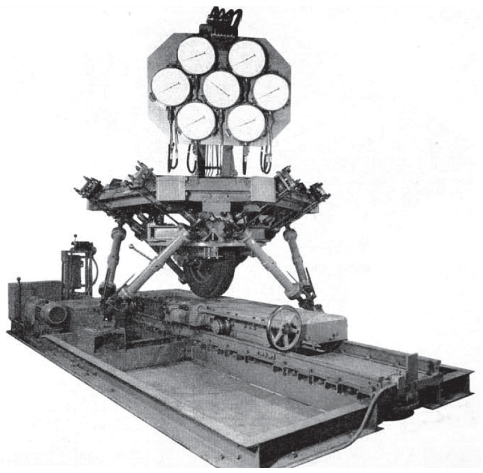


Figure 1.3 – Original Gough platform

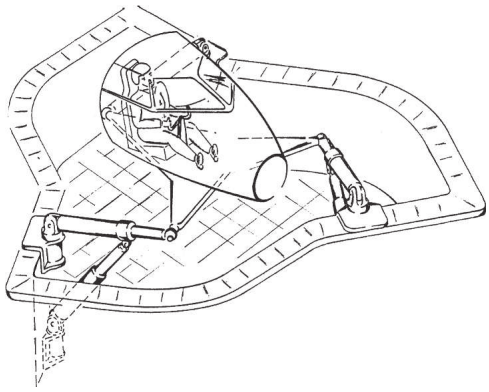


Figure 1.4 – Stewart's flight simulator

The above discussion enumerates some prototypes that have marked the history of PKMs. It is not, however, intended to be an exhaustive list of all existing ones.

1.2.2 Some successful examples

Due to the growing interest in PKMs, both in industry and academia, several PKMS prototypes with impressive features have been proposed in the literature. Most of these mechanisms were based on Gough platform and, later, on the Delta robot. However, only

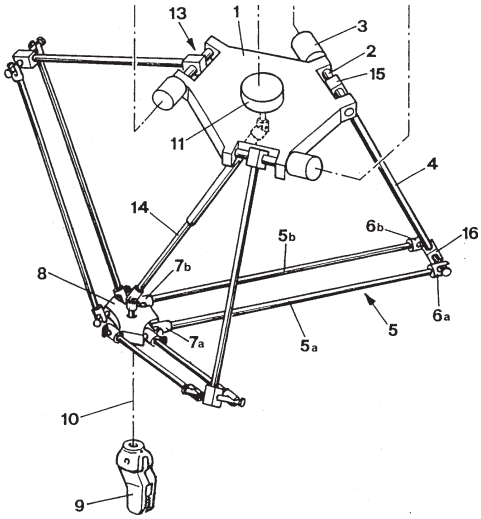


Figure 1.5 – Technical drawing of the original Delta robot (US Patent 4,976,582) [Clavel, 1985]

few of these prototypes have made their way to industry up to the commercialization. In the following, some examples of parallel robots widely adopted in industry are described.

Gough platform

As described earlier, the first parallel manipulator ever industrialized is the Gough platform. This PKM, belonging to the 6-DOF haxapods family, is considered as one of the most studied parallel platforms in academia. It has six, hydraulically actuated, extensible links connecting the base to the moving platform. The variation of the links' lengths induce the change in the position and orientation of the moving platform. Since the actuators in Gough platform are located on the links, its dynamic performance is relatively limited compared to lightweight PKMs, where the actuators are located on the fixed base. However, one of the main motivations behind its development is to split up the weight of the carried payload over the six kinematic chains. Consequently, such mechanisms are capable of moving heavy payloads with accurate positioning.

The Gough platform has been used in various industrial applications such as tire testing, flight simulation, positioning device, machine-tools, etc (cf. Figure (1.6)).

(a) Flight simulator (Lufthansa AG)

(b) Positioning device (Symétrie France)

Figure 1.6 – Some applications of the Gough platform

Delta robot

The Delta robot [Clavel, 1985], developed by professor Clavel at the EPFL in the early 80's, was the first lightweight PKM. It has only three DOFs, meaning less mobility than the usual six DOFs commonly used in Gough platform. The main idea of the Delta robot is to

use three lightweight parallelograms that constrain the motion of the end-effector to the three translational DOFs. Furthermore, the robot is quipped with three revolute actuators being located on the fixed base which enables very high accelerations. It should be noted however that the original Delta robot as proposed by professor Clavel has an extra rotational DOF generated by a central telescopic arm. The Delta robot is known to be the perfect candidate for pick-and-place operations of light objects. It is typically used in packaging industry, pharmaceutical industry, surgical applications and in assembly of electronic components. Figure 1.7(a) shows the IRB 340 FlexPicker, an industrial PKM based on the Delta design developed and commercialized by ABB Flexible Automation. This fast PKM is intended to be used for food packaging as well as pharmaceutical and electronic industries. Figure 1.7(b) shows the SurgiScope, a Delta robot designed to carry a heavy microscope of 20 kg.

(a) IRB 340 FlexPicker (ABB Flexible Automation)

(b) SurgiScope: a Delta robot carrying a microscope (Elekta)

Figure 1.7 – Some applications of the Delta robot

Quattro robot

Adept Quattro is the industrial commercial version of a fast PKM prototype called Par4, a 4-DOF PKM (cf. Figure 1.9(a)) developed at LIRMM dedicated to high-speed pick-and-place applications [Nabat et al., 2005]. Its architecture is inspired from the Delta robot with the peculiarity of having four identical kinematic chains instead of three and an articulated traveling plate. Unlike the Delta robot where the rotation of the moving platform (traveling plate) is provided by a telescopic arm, the Par4 is equipped with an articulated traveling

plate (cf. Figure 1.9(b)) that improves the stiffness of the traveling plate at the workspace extremities [Nabat et al., 2005]. Furthermore, the parameters of Par4 are optimized in a way to have the best kinematic and dynamic performance with respect to a typical industrial pick-and-place trajectory with specified parameters (cf. Figure 1.8).

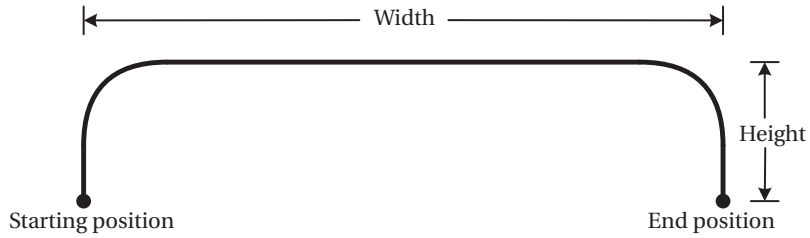
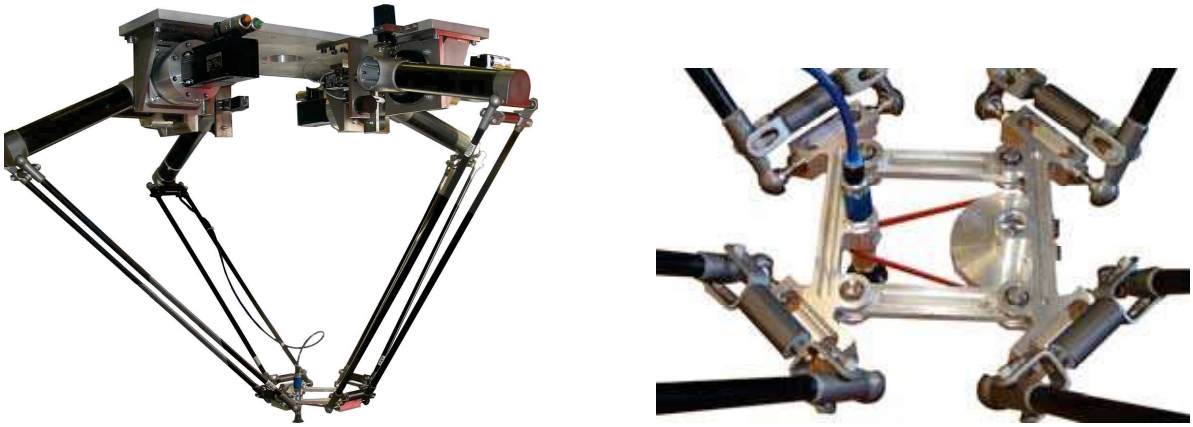


Figure 1.8 – A typical pick-and-place trajectory

The Par4 prototype has been industrialized and is being commercialized by Adept Technology, Inc. under the name of Quattro (cf. Figure 1.10). It is considered as the fastest industrial parallel manipulator in the world. Similar to Par4, the Quattro robot is equipped with an articulated traveling plate shown in 1.10(b).



(a) Par4 robot (LIRMM) (b) Traveling plate of Par4

Figure 1.9 – Par4 robot and its articulated traveling plate

1.2.3 Parallel versus serial manipulators

Parallel manipulators hold several differences compared to their conventional serial counterparts. Although it is often reported that parallel manipulators offer several advan-

(a) Quattro robot (Adept Technology, Inc.) (b) Traveling plate of Quattro

Figure 1.10 – Quattro robot and its articulated traveling plate

tages over serial ones, it should be mentioned that each one of them is dedicated to a particular class of applications. Therefore, it would be preferable to say that parallel manipulators offer various advantages over serial ones for certain applications [Patel and George, 2012].

Table 1.1 – Main characteristics of serial and parallel manipulators

	Type of manipulator	
Property	**Parallel manipulators**	**Serial manipulators**
Structure type	Closed-loop	Open-loop
Actuators' location	Near or on the fixed base	On the joints
End-effector	Traveling plate or moving platform	Gripper
Natural description	In joint space	In Cartesian space
Preferred property	Stiffness	Dexterity
Preferred application	Precise positioning	Large motion
Singularity	Static	Kinematic
Inertia forces and stiffness	Less and high respectively	High and less respectively
Foreword kinematics	Difficult and complex	Straightforward and unique
Inverse kinematics	Straightforward and unique	Difficult and complex

Table 1.1 shows the main properties of both serial and parallel manipulators. The most notable advantages can be summarized as follows [Corbel, 2008]

- Significantly higher load capacity to robot mass ratio.

- High structural stiffness due to the closed-chains structure.
- Great dynamic performance due to a low moving mass.
- Better tracking accuracy (higher precision).

However, these advantages do not come without drawbacks. The main drawbacks of parallel manipulators can be summarized in the following [Corbel, 2008]

- Smaller workspace due to links interference.
- Complex forward kinematics (sometimes there is no unique solution).
- Large number of possible existing topologies (difficult to choose the best one).

Several academic works were conducted to overcome the above limitations and improve the acceptance of parallel manipulators in industry. The proposed solutions include, among others, actuation redundancy [Merlet, 2006], articulated traveling plate [Pierrot et al., 2009] and optimization of the manipulator's parameters [Yuehua et al., 2008].

1.3 Some examples of typical applications of PKMs

As discussed earlier, the first two PKMs prototypes have never been built. The first prototype, designed by James E. Gwinnett was intended to be used as a dynamic cinema. This invention was expected to be combined with a moving picture to add a feeling of action to the viewer. The second prototype, developed by Willard L.V. Pollard, was expected to serve as a spray-painting machine. The manipulator's end-effector was intended to hold a spray gun in order to control its position and orientation with respect to the surface to be painted. The main reason why these PKMs have not been built is their high complexity for the industry in that time.

However, with the increasing advances of industrial techniques, several PKMs have been built and commercialized in the last few years. Hereafter we enumerate some industrial applications that have been proposed to be accomplished by PKMs. Some prototypes were already built and commercialized, while the other ones are still in conceptual design phase.

1.3.1 Machine tools

A machine tool is a machine for shaping and changing the form of rigid materials such as metal, plastic, wood, etc. Traditional machine tools are based on serial manipulators where each link supports all the following ones and contribute to their movements. Consequently, to provide the required stiffness and dynamic performance for the end-effector, the moving parts should be heavy and the actuation should be powerful enough.

Parallel manipulators offer several advantages, compared to conventional serial mechanisms in terms of stiffness, accuracy, acceleration, etc. Since the recent trend in industrial

Figure 1.11 – Toyoda HexaM PKM-based machine tool (US Patent 5,715,729) [Toyama et al., 1995]

machining is toward high-speed machining, parallel manipulators are being progressively adopted to develop this kind of machines. Figure 1.11 shows a PKM-based machine tool having six DOFs developed by Toyoda Inc.

Parallel machine tools can be classified into three main categories according to their actuation type [Zhang, 2010]

1. Parallel machine tools with prismatic joints and variable leg lengths (e.g. Octahedral Hexapod [Lauffer et al., 1996]).
2. Parallel machine tools with revolute joints and fixed leg lengths (e.g. Hexa [Pierrot et al., 1991]).
3. Parallel machine tools with linear actuators and fixed leg lengths (the actuators move along linear guideways) (e.g. Hexaglide [Honegger et al., 1997]).

1.3.2 Laser cutting

Laser cutting is the process of using high-powered laser beam directed towards a work-piece in order to cut material. A major advantage of laser cutting compared to competitive technologies is the high-quality of the resulting cut product.

Thanks to their advantageous structural properties, PKMs are being promoted as the best alternative to conventional machine tools for the handling of the laser cutting head. Indeed, among the main requirements of high-quality laser cutting we can mention high accelerations required for sharp edge cutting and high-accuracy which is mainly related to the structural stiffness. Consequently, PKMs seem to be a good candidate for the handling of the laser cutting head.

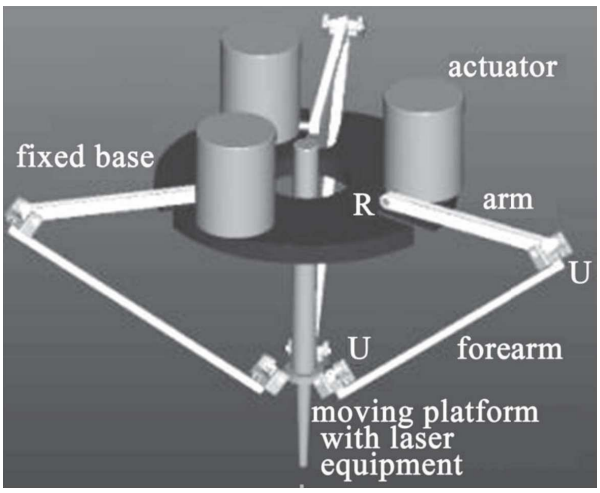


Figure 1.12 – PKM-based laser machine prototype [Bruzzone et al., 2002]

Although the idea of using PKMs for laser cutting tasks seems to be very promising, it has not been addressed enough in the literature. One of the few work that could be mentioned regarding this subject is the proposed prototype in [Bruzzone et al., 2002]. In this work, a mechanical structure based on the Delta robot was proposed as a solution for the laser cutting head. A conceptual design of the proposed PKM-based laser cutting machine in this work is depicted in Figure 1.12.

1.3.3 High-speed pick and place

Pick-and-place can be defined as the process of picking up an object from one location and dropping it off in another one. Depending on the application, a certain accuracy is required, mainly at the stop points (start and end points). A typical pick and place trajectory (known as Adept cycle) is illustrated in Figure 1.8. Thanks to their impressive dynamic performance, PKMs are the perfect candidate for pick-and-place tasks.

According to [Merlet, 2006], the first commercialized PKM is a 6-DOF parallel manipulator called Gadfly, designed by Marconi Company Ltd. The Gadfly was intended to be used for the assembly of electronic components, which is a pick-and-place application. Probably the most successful pick-and-place parallel robots are the Delta and Quattro robots. Figure 1.13 shows an industrial bakery powered by Line-Placer Delta robots (fabricated by Demaurex company) performing pick-and-place tasks.

1.3.4 Medical applications

Though PKMs are mostly used in industry, some researchers investigated the possibility of using them in medical applications. Indeed, PKMs hold many advantages that make

Figure 1.13 – Industrial bakery powered by parallel robots

them perfect candidates for some specific medical procedures. For example, in [Li and Xu, 2007], the authors proposed to use a Delta-like robot for cardiopulmonary resuscitation (CPR) on patients undergoing cardiac arrest. The motivation behind robotizing this medical intervention is to avoid eventual virus transfer due to contact with the infected patients. Moreover, performing CPR is energy consuming and requires a lot of effort from the doctors, hence, the benefits of robotizing such task. A conceptual design of the proposed PKM-based CPR system is illustrated in Figure 1.14.

Another illustration of the use of PKMs for robot-assisted surgery (RAS) can be found in [Shoham et al., 2003]. In this work, the authors proposed to directly mount a miniature PKM on the patient's anatomy in order to perform a surgical procedure. The goal of the PKM is to accurately position an instrument tip with a specific orientation. Figure 1.15 illustrates this concept where a proposed 6-DOF MiniAture Robot for Surgery (MARS) PKM is involved in a spinal surgical procedure.

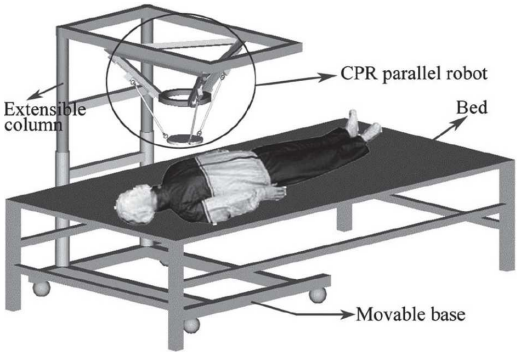


Figure 1.14 – Design of a CPR PKM

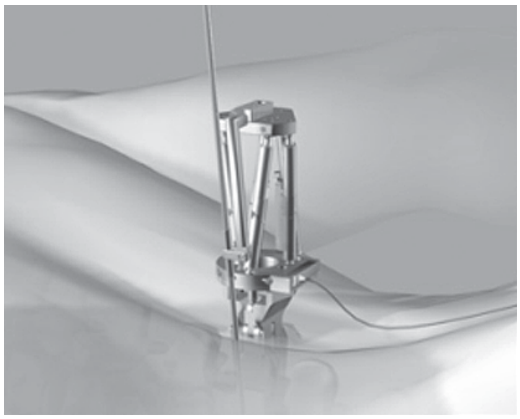


Figure 1.15 – PKM in a spinal procedure

1.3.5 Haptic devices

Haptic devices are recent mechanisms providing force feedback for the user to feel the environment he is interacting with. It is a means to provide an additional knowledge about the environment for some tasks where visual feedback is not enough. Haptic devices can be found for example in RAS and teleoperation. Inherent characteristics of PKMs enable them to be the best candidates for recent sophisticated haptic devices. Indeed, most of the important features for high-performance haptic devices can naturally be found in PKMs. Such features include low inertia, high stiffness and low friction.

Figure 1.16 illustrates a PKM-based haptic device with spherical geometry called SHaDe (acronym for **S**pherical **Ha**ptic **De**vice) [Birglen et al., 2002]. This device has the particularity of having a fixed point with unlimited possible rotation around it, allowing a very large workspace [Bonev, 2014]. The Omega6 PKM-based haptic device, depicted in Figure 1.17, is one of the most advanced force-feedback interfaces. It provides 3D force feedback with decoupled translations and rotations.

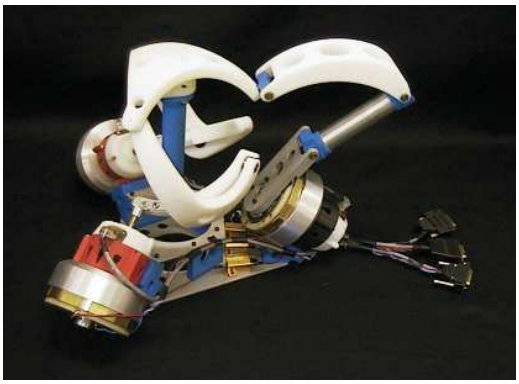


Figure 1.16 – Shade: a 3-DOF PKM-based haptic device

Figure 1.17 – Omega.6: a 6-DOF parallel haptic device

1.4 The ANR-ARROW joint project

The current thesis has been conducted within a French National Research Agency (ANR) project called ARROW.

ARROW is an acronym standing for **A**ccurate and **R**apid **R**obots with large **O**perational **W**orkspace. It is a national joint project aiming at conducting academic research to investigate possible solutions enabling PKMs to be both fast and accurate. ARROW is the successor of another ANR project dedicated for PKMs called Objective 100g, which has as objective to develop a PKM capable of reaching 100 g (i.e. 1000 m/s^2) of maximum acceleration !

The ultimate goal of ARROW is to design and build parallel manipulators capable of reaching high accelerations (20 g) while keeping very low tracking errors (20 μm). To that aim, the project is structured around two major research axes: optimal design and advanced control.

1.4.1 Project's consortium

Since the challenges and problematics that will be encountered in the ARROW project are numerous including various research topics, a collaboration of different experts in the domain of robotics and automatic control is required. The following partners are directly involved in the consortium of the project.

- **IRCCyN** - Robotics team: The research activities in the Institute of Research in Communications and Cybernetics of Nantes include automation, signal and image processing, robotics, etc. One of the main objectives of the robotics team at IRCCYN that fits with the context of the ARROW project is the optimal design, modeling and identification of industrial robots.
- **LIRMM** - Robotics department - Dexter Team: The research activities of the Dexter team within the Laboratory of computer science, Robotics and Microelectronics of Montpellier essentially include design and control of efficient medical and industrial robots. The Dexter team have been working for several years on national and European projects concerning parallel machines, such as ECHORD-PRADA, ANR-CoGiRo and ANR-Objectif 100g. Throughout the years, the Dexter team has acquired sound skills and a good expertise in the field of parallel manipulators. The conducted research efforts of the team culminated in the development of the world's fastest industrial PKM, namely the Adpet Quattro. This industrial robot can reach a peak acceleration of $200 m/s^2$ while carrying a payload of up to 2 kg.
- **Tecnalia** is a non-profit research development and innovation group resulting from the merging of several research organizations. Tecnalia, whose headquarter is located in Spain, is the fifth largest independent research entity in Europe. The main focus of the Tecnalia foundation activities is the applied research on innovative technologies and the transfer of technology (from academia to industry). ARROW is an interdisciplinary project, it requires strong skills at all levels. Since an important part of ARROW involves realization and testing of the developed prototypes, the main contribution brought by Tecnalia within the project is to conduct and follow the realization of the developed PKM prototypes.

1.4.2 Objectives of the project

The ARROW project addresses mainly the possible solutions to improve the accuracy of industrial robots operating at high speeds. More precisely, the ultimate goal is the de-

sign and construction of a parallel robotic manipulator able to operate at 20g of acceleration while keeping an absolute accuracy of 20μm in trajectory following as well as at the stop-points in pick-and-place applications. While this goal might seem very ambitious, the experience of the different partners of the project in this field points out that several improvements can be carried out in order to reach this goal. In terms of scientific development, the performance gain can be achieved through improving the design methodologies, the modeling and identification techniques and the advanced control schemes.

A thorough study of the existing methods in the literature reveals that various scientific topics are still to be investigated in order to improve the performance of fast parallel manipulators. Among these topics we can mention:

- Defining performance evaluation criteria useful for the choice of the architecture of the robot.
- Improving the modeling techniques of flexibilities in multi-body systems.
- Propose optimal solutions in terms of trajectory generation to minimize the vibration phenomena in the mechanism.
- Developing advanced control techniques suitable for high-speed mechanisms.

It is now clear that the ARROW project involves various scientific topics that require cooperation and synergies between academic and industrial partners.

1.4.3 Organization of the project

In order to enhance the performance of PKMs, two main research axes are considered within the ARROW project

- **Axis 1:** Optimal design of rapid and accurate robots.
- **Axis 2:** Advanced control of rapid and accurate robots.

In this framework, five main tasks are considered, each task covers a specific research topic. The tasks are not completely independent, in other words, the success of one task may not be possible without the outcomes of another one. In the sequel, a brief description of each task is provided

- **Task 1 - Project coordination:** this task is responsible for the coordination and the synchronization of all the remaining four tasks. Hence, it interacts with all the remaining tasks and aims to supervise and ensure the good progress of the project.
- **Task 2 - Optimal design:** this task falls within the first, above mentioned, axis of research. It deals with design considerations of the developed PKMs and the optimization of their mechanical structures while considering all the specifications of the project. In other words, defining performance criteria to be satisfied in order to fulfill the specified requirements in terms of accuracy and accelerations.

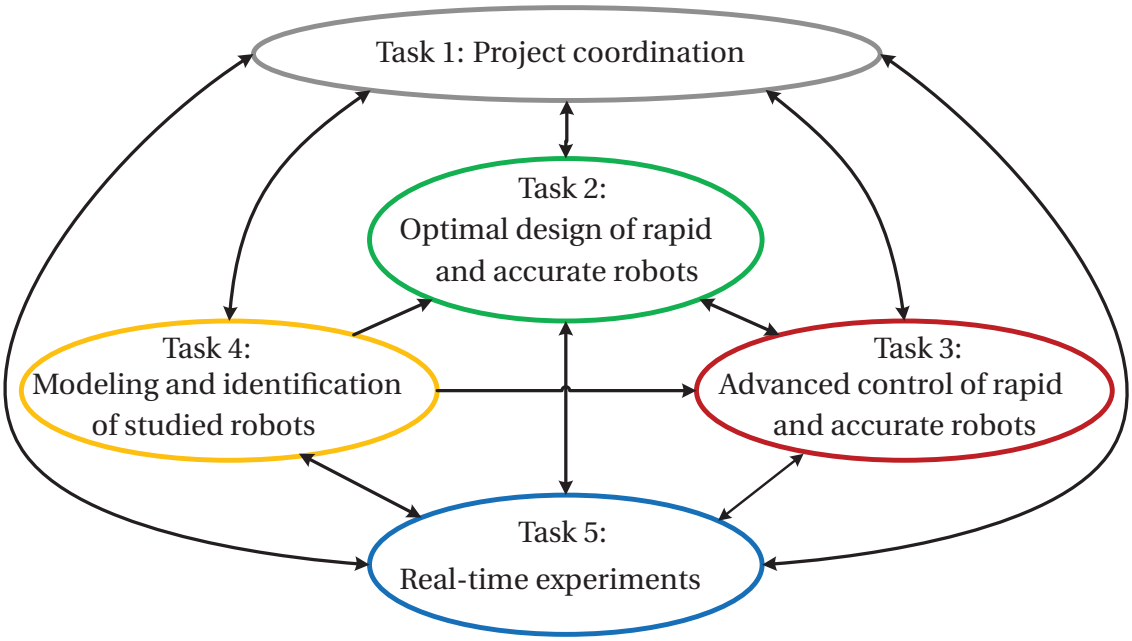


Figure 1.18 – ARROW Project: interaction between the different tasks

- **Task 3 - Advanced control:** the current thesis falls entirely in this task which belong to the second research axis. It consists of developing advanced control strategies able to perform accurate trajectory tracking while dealing with the various phenomena to be encountered in order to ensure the desired performance.
- **Task 4 - Modeling and identification:** the outcomes of this task, dealing with modeling and identification of the developed prototypes, are required for tasks 2 and 3. As it will be investigated in the next chapter, modeling has an important role for efficient control. Furthermore, modeling is necessary to evaluate the developed performance criteria in task 2.
- **Task 5 - Real-time experiments:** as previously explained, one of the main goals of the ARROW project is to fabricate the developed PKMs prototypes and evaluate their performance. Indeed, real-time experiments of the proposed control schemes within this thesis (task 3) are also explicitly required in the set of specifications of the ARROW project.

The organization of the ARROW project and its different tasks are illustrated in Figure 1.18.

1.4.4 Developed prototypes

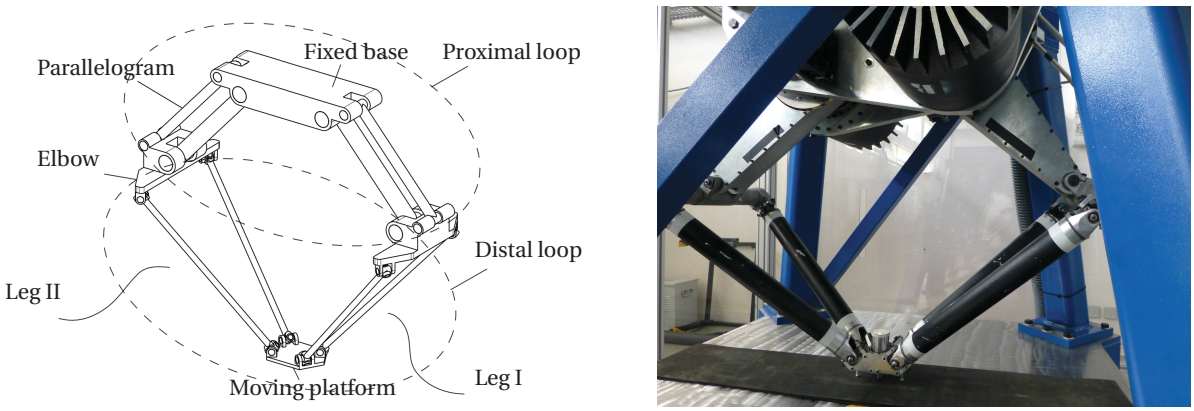
As it was discussed earlier, one of the main objectives of the ARROW project is to design fast and accurate PKMs with large operational workspace. Moreover, one main goal is to fabricate the developed prototypes and assess their capabilities given the constraints and

limitations usually encountered in practice. In this context, two PKMs prototypes were developed. The first PKM is the IRSBot-2 (cf. Figure 1.19), developed at IRCCYN, dedicated to high-speed pick-and-place operations. The second developed prototype is ARROW robot and is intended to be used for machining tasks. In the following, a brief description of the IRSBot-2 and ARROW parallel manipulators is provided.

IRSBot-2: A 2-DOF pick-and-place PKM

The IRSBot-2 (an acronym for **IR**CCyN **S**patial ro**Bot** with **2** DOFs) is a 2-DOF PKM developed at IRCCyN within the ARROW project. The IRSBot-2 is intended to be used for high-speed pick-and-place tasks where only two translations are required (see Figure 1.8 for instance).

The mechanical structure of IRSBot-2 is illustrated in Figure 1.19(a). The moving platform is linked to the base through two identical kinematic chains. Each chain is composed of a proximal module and a distal module. Each chain is actuated by one direct drive actuator mounted on the fixed base allowing the rotational motion of the proximal link with respect to the actuator's axis. The distal links are linked to the to the proximal ones and to the platform by means of revolute joints.



(a) View of the mechanical structure

(b) The fabricated IRSBot-2

Figure 1.19 – The IRSBot-2 parallel robot (IRCCyN)

The IRSBot-2 was designed with the specifications of the ARROW project in mind. Among these specifications we can mention the most significant:

- Maximum acceleration: 20 g.
- Pick-and-place trajectory dimensions: 25 mm × 300 mm.
- Cycle duration: 200 ms.
- Number of cycles per minute: 300 cycle/min.

- Repeatability: 20 μm.

In order to fulfill the above specifications, the following design guidelines have been considered for the synthesis of the architecture of IRSBot-2:

- The actuators are located on the fixed base to minimize the mass and inertia in motion,
- the use of revolute actuators, less expensive and faster than translational ones,
- only two kinematic chains are considered in order to increase its operational workspace, minimize the mass in motion and reduce the mechanism complexity,
- the kinematic chains are identical in order to facilitate the fabrication process,
- avoid the use of spherical joints to ease the manufacturing of the robot.

The IRSBot-2 overcomes its similar counterparts (see Par2 robot [Company et al., 2011] for instance) in terms of mass in motion, stiffness and workspace size. Moreover, its design parameters have been optimized such that the robot never encounters any parallel singularities [Germain et al., 2013].

The first fabricated prototype of the IRSBot-2 is shown in Figure 1.19(b) For further details about its design, modeling and performance analysis, the reader is referred to [Germain, 2013].

ARROW: A 5-DOF machining PKM

As discussed earlier, the ARROW project involves the design of two PKMs, designed for two different applications. The ARROW robot is the parallel manipulator prototype developed by LIRMM within the ARROW project (the robot holds the same name as the project) intended to be used for machining tasks. It consists of a 5-DOF (three translations and two rotations) robot composed of a PKM module that provides four DOFs and a turntable providing one rotational DOF.

The main novelty of ARROW robot lies in its design and dual-criteria optimization [Shayya et al., 2014]. The overall mechanism is actuated by six linear actuators, all aligned along the same direction. The set of actuators can be considered to be split up into two sets, each set consists of three actuators supplying the required actuation to the moving platform. The moving platform is linked to the fixed base through two pairs of simple arms and one pair of parallelograms. The pair of parallelograms constrains the rotation of the moving platform around the vertical axis. The simple arms, in addition to the parallelograms, contribute to the positioning of the moving platform in the space.

The ARROW robot have the following main advantageous features:

- Large workspace, especially along the sliders' direction.
- Absence of singularities of all types inside its workspace.
- Homogeneous dynamic performance over its workspace.

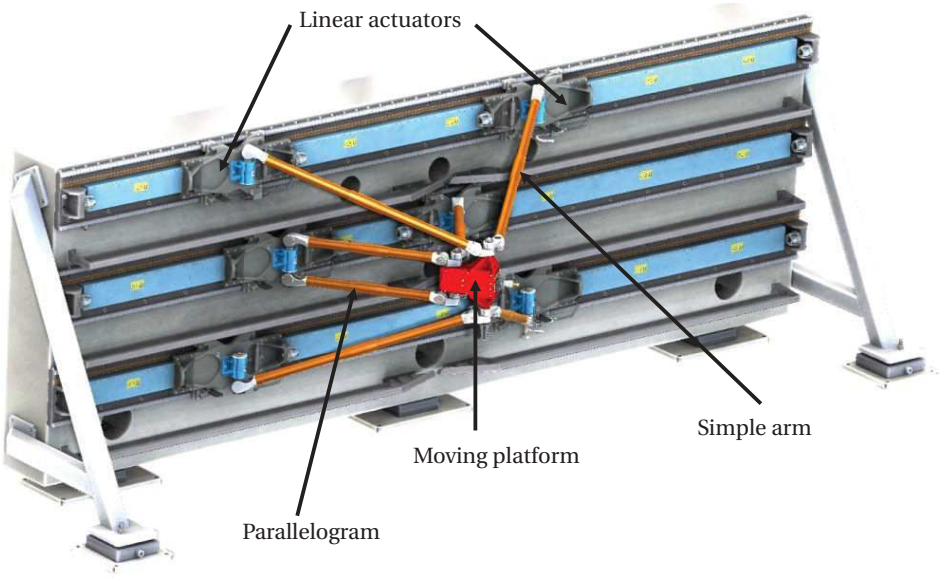


Figure 1.20 – CAD view of ARROW robot

Moreover, its design parameters have been optimized using novel dual criteria that consider both precision and dynamics simultaneously [Shayya et al., 2014]. For more details about the theoretical background and design synthesis behind the development of the ARROW robot, the reader is referred to [Shayya, 2015].

1.5 Control problem formulation

In order to fulfill the requirements of the ARROW project, the development of advanced control schemes that consider every aspect of the manipulator including its dynamics and environment is mandatory. Since the dynamics of parallel manipulators are known with their highly nonlinear nature, advanced nonlinear control strategies have to be considered to reach the desired performance. Furthermore, actuation redundancy, variation of the dynamics and the abundant uncertainties make the control task even more challenging.

1.5.1 Complexity of the dynamics

PKMs are known by their highly nonlinear dynamics mainly emerging from their closed-kinematic-chains structure. Moreover, their nonlinearities increase considerably when operating at high accelerations. While linear systems are well established in control theory, nonlinear systems are known to be challenging and difficult to be dealt with. Furthermore, the closed-kinematic-chains structure gives rise to coupled dynamics between the different kinematic chains. Consequently, any tracking failure on a specific actuator

will propagate to the others resulting in a loss of tracking and possible damages to the mechanical structure. As a result, control of parallel manipulators is often described in the literature as a very challenging task.

1.5.2 Actuation redundancy

Redundantly actuated PKMs are characterized by their number of actuators greater than their number of degrees of freedom [Merlet, 2006]. In other words, parallel manipulators with actuation redundancy are characterized by the following property:

> " when an external force is exerted on the moving platform of a redundantly actuated PKM, there exist an infinite possible combinations of actuators' torques (forces) that can counterbalance this force [Corbel, 2008]."

Since actuation redundancy means that there are more actuation sources than degrees of freedom, the notion of degree of actuation redundancy is introduced. It consists of a positive integer computed as the difference between the number of actuators of the PKM and its number of degrees of freedom.

Actuation redundancy can be obtained by three main solutions [Ganovski, 2007]:

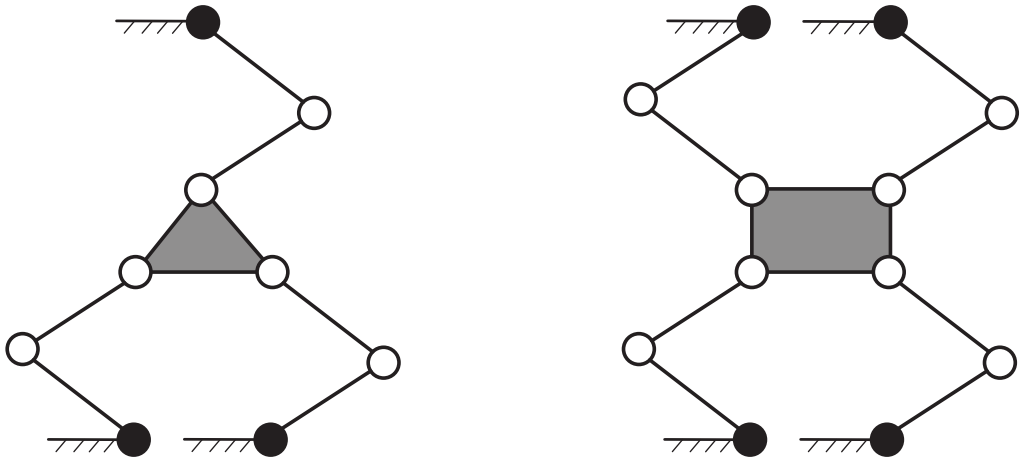
- Actuation of the passive joints of the mechanism.
- Introducing additional actuated kinematic chains (often identical to the other chains).
- A combination of the two previous solutions.

An illustrative example of actuation redundancy for a 3-DOF PKM through the addition of an actuated kinematic chain is illustrated in Figure 1.21. In this example, the actuated joints are represented by black circles.

Actuation redundancy offers several advantages for parallel manipulators in terms of versatility and performance [Ganovski, 2007]. These advantages include:

- Elimination of singularities inside the workspace.
- Increasing the dynamic capabilities of the PKM.
- Homogenizing of the dynamic performance throughout the workspace.

Furthermore, the possession of more inputs than required gives rise to the possibility of planning secondary tasks in addition to the trajectory tracking. Such possibilities include stiffness modulation, fault tolerance, singularity cancellation, backlash avoidance, etc. However, from control point of view, actuation redundancy brings some challenges and additional difficulties to the surface. First, the addition of more actuation sources and kinematic chains leads to more kinematic constraints, and hence, the complexity of the dynamics is increased. Moreover, actuation redundancy induce unavoidable internal forces that may harm the mechanical structure. These internal forces be dealt with by means of control to minimize their effects.



(a) Non-redundant 3-DOF PKM (b) Redudnantly actuated 3-DOF PKM

Figure 1.21 – Illustration of actuation redundancy through additional kinematic chain

1.5.3 Uncertainties and variations of the dynamics

Uncertainties are abundant in the case of PKMs. These uncertainties may be originating from the PKM itself or from the environment it is interacting with. Examples of sources of uncertainties include

- The wear of the parts of the robot leading to a change in the dynamic parameters of the robot.
- Uncertainties in the modeling due to dissimilarities between CAO design and the fabricated robot.
- Non-modelled phenomena and simplifying hypotheses when establishing the dynamic model of the PKM resulting into inaccurate dynamics.
- External disturbances resulting from the interaction with the environment (contact with the environment, payload handing, etc.).

Such uncertainties and variations result in a complex and unexpected behavior of the dynamics of the robot. If the implemented control scheme does not take into consideration all the uncertainties during its design phase, it may lead to undesired behavior, loss of tracking and even instability of the controlled system.

1.6 Objectives de the thesis

The objectives of this thesis falls into **Task 3** of the ARROW project presented in section 1.4. This task concerns the design of advanced control strategies allowing to improve the

tracking performance of PKMs. It is undeniable that rapidity and accuracy are very important and advantageous features of parallel manipulators. However, it can be noticed that these two features are often treated separately in the literature. This is mainly due to the existing dilemma between them, not only in parallel manipulators, but in all dynamic systems. The ARROW project considers both aspects simultaneously. In other words, the proposed PKMs within the project have to fulfill the defined accuracy and rapidity specifications simultaneously. To that aim, the following objectives within this thesis are considered:

- Development of efficient control schemes to be implemented on PKMs for the tracking of desired trajectories. A special care is paid to the tracking accuracy of the moving platform of the robot as it is vital for many PKM applications (pick-and-place, machining, etc.). To that end, the proposed control solutions should consider every aspect of parallel manipulators in terms of complexity and uncertainty in the dynamics.
- Implement and validate the proposed control schemes through numerical simulations at first. This is a mandatory step for every control design to ensure its stability and to evaluate its performance before proceeding to real-time experiments on the real robot.
- Implement the proposed control solutions in real-time experiments on available PKMs. Various parallel robots are considered for the implementation including redundantly actuated and non-redundant robots.

1.7 Main contributions of the thesis

The contributions of this thesis consist in the design, analysis and implementation of advanced control schemes for parallel manipulators operating at high accelerations. Given the nonlinear, uncertain and time-varying nature of their dynamics, adaptive control schemes were chosen as the main ground basis for our developed control solutions. In particular, the following key points summarizes the man contributions of this thesis

- Extending an existing adaptive controller based on feedforward compensation by replacing its constant feedback gains with nonlinear time-varying ones. This contribution is motivated by the superiority of the nonlinear gains that has widely been demonstrated in the literature through several research works.
- Developing a novel control scheme combining both robust feedback and adaptive control. The robust feedback strategy in question is called RISE control (Robust Integral Sign of the Error) and it has never been applied to PKMs before. The combination of adaptive control and RISE is motivated by the aim to improve the tracking performance of the controlled system.
- The appropriate adaptation and application of the recently developed $\mathcal{L}_1$ adaptive control to PKMs. This control strategy has not been applied on parallel manipulators yet

since it stills in its early development stage. $\mathcal{L}_1$ adaptive control has particularly attracted our attention thanks to its robustness and adaptation.

- Enhancing $\mathcal{L}_1$ adaptive control, originally independent from the dynamic model, by including a model-based compensation term. First, a nominal feedforward has been added to the control architecture to perceive the relevance of this proposed extension. Then, the included dynamics are endowed by adaptation capabilities in order to deal with variations of the dynamics.
- Real-time experimental validation of all the proposed control solutions on available prototypes, including the ARROW robot. The performance of each proposed control scheme and its robustness are evaluated through multiple operating scenarios.

1.8 Conclusion

This chapter introduced the context of this thesis as well as the challenges that should be encountered when dealing with control of parallel manipulators. The main goals of this thesis, being conducted within the framework of a French national research project, is to develop efficient control schemes for parallel manipulators. These robots are known by their complex dynamics that could be uncertain, time-varying and subject to various sources of external disturbances. Hence, in order to fulfill the desired specifications of the ARROW project in terms of tracking performance, the developed control schemes have to consider these challenges and to deal with them in real-time.

The next chapter will present a state of the art on control strategies proposed in the literature for PKMs.

CHAPTER

2 State of the art

Contents

2.1 Introduction

It is undeniable that control plays an important role in improving the performance of PKMs and contribute in exploiting their effective potential. Since this thesis aims at proposing advanced controllers to fulfill the requirements of the ANR-ARROW project, it is vital to give first a survey on the most relevant control strategies proposed in the literature. This will allow us to compare the proposed solutions with the existing control strategies and build a solid base from where we start the development of our proposed control schemes.

In this chapter we provide an overview about the existing solutions proposed in the literature regarding control of PKMs. It is not intended to be an exhaustive list but it provides a good overview about existing control strategies in the field. But first, we briefly recall the equations of the dynamic modeling of PKMs and how they can be obtained.

2.2 Dynamic modeling of PKMs

Dynamic modeling plays an essential role whenever efficient control is considered. First, a dynamic model is required for simulation purposes before proceeding to the real-time implementation of any controller. Numerical simulations allow to predict the behavior of the system under control in order to ensure the stability of the closed-loop and the respect of physical limitations of the system. Moreover, most modern control strategies rely on the dynamic model of the plant to achieve the specified control objective. Indeed, most of the recent controllers include the available knowledge about the dynamics of the system in order to improve the closed-loop performance [Spong and Vidyasagar, 1989].

Several methodologies and techniques exist in the literature in order to establish the dynamic model of PKMs [Yiu et al., 2001]. The classical method first considers the dynamics of the equivalent tree structure, then it considers the constraints by using the Lagrange multipliers or d'Alembert's principle [Cheng et al., 2003]. Other strategies include the virtual work's principle [Tsai, 2000], Lagrange formalism [Abdellatif and Heimann, 2009], Newton-Euler equations [Dasgupta and Choudhury, 1999], etc. One interesting and relatively recent approach is that proposed by [Nabat, 2007]. This method allows to obtain a fairly accurate and simple dynamic model for PKMs adequate for real-time experiments. Throughout this thesis, we consider that the dynamics of the PKM is obtained via the proposed approach in [Nabat, 2007].

To establish the dynamic model of PKMs according to [Nabat, 2007], it is common to assume some hypotheses. These hypotheses help in simplifying the establishment of a simplified dynamic model suitable for real-time implementations. While it's possible to compute a more complete dynamic model, this latter would require complex computations and it would not be possible to evaluate it in real-time. The following assumptions are often considered

- The rotational inertia of the forearms of the manipulator is neglected in the dynamic model. This hypothesis is justified by the fact that the forearms of most Delta-like robots are made with lightweight materials.
- The mass of each forearm is split up into two point-masses, located at the end-points of the forearm. Consequently, the mass contribution of the forearms in the dynamics of the manipulator is considered with the arms and the end-effector (a traveling plate). The distribution of the arms' masses, considering this hypothesis, is illustrated in Figure 2.1.
- Gravitational forces can be neglected if very high motion accelerations are considered. In fact, the gravitational term can be omitted because its contribution to the dynamics is negligible compared to the other effects.

According to [Nabat, 2007], the dynamics of parallel manipulators can be obtained by analyzing the dynamics of the actuators and the traveling plate separately. Then, the complete dynamic model is obtained by summing-up the two equations of motion. As mentioned previously, the forearms are represented by two equivalent point-masses (cf. Figure

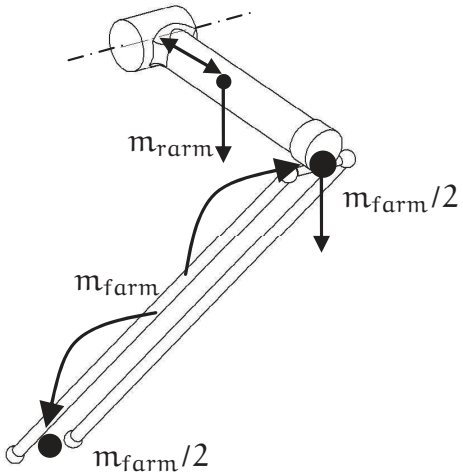


Figure 2.1 – Illustration of the masses' distribution

2.1). Hence, the dynamic model is obtained by considering the effects of the motions of the traveling plate and the actuators. Accordingly, for a parallel manipulator having n actuators we can write

$$\Gamma(t) = \Gamma_{act}(t) + \Gamma_{tp}(t) \tag{2.1}$$

where $\Gamma_{act}(t) \in \mathbb{R}^n$ is the actuators' torques (efforts in the case of linear actuators) and $\Gamma_{tp}(t) \in \mathbb{R}^n$ stands out for the forces resulting from the movements of the traveling plate.

- The first term $\Gamma_{act}(t)$, which represents the contribution of the actuation, consists of the combination of the following components:

  The dynamics of the actuators.

  The dynamics of the rear arms.

  The dynamics of the half-masses corresponding to the forearm.

- The second term in (2.1), $\Gamma_{tp}(t)$, is the contribution of the traveling plate to the dynamics of the PKM. This term depends on the pose and acceleration of the traveling plate as well as an eventual payload and possible external disturbances (due to the contact with the environment).

Following the above partitioning scheme of the dynamics, two formulations are possible. The first is to consider that the dynamics resulting from the actuation are acting on the traveling plate as external forces. Then, the formulation of the dynamics is written in the traveling plate's coordinates (Cartesian, operational or task space). However, if the forces resulting from the movements of the traveling plate are transferred to the forearms, and then to the actuators, we say that the dynamics is written in joint space. The Cartesian space dynamic model is suitable for Cartesian-space control while a joint-space dynamic model is preferred for joint space control.

Throughout this thesis, we consider that the dynamic model of any m-DOF PKM having n actuators (such that $n \geq m$), can be put into the standard joint-space form as follows [Siciliano et al., 2009]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + \Gamma_d(t) = \Gamma(t) \tag{2.2}$$

where

$M(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix.

$C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is the centrifugal and Coriolis matrix.

$G(q) \in \mathbb{R}^{n}$ is the gravitational forces vector.

$\Gamma_d(t) \in \mathbb{R}^{n}$ accounts for possible external disturbances and non-modeled dynamics.

$\Gamma(t) \in \mathbb{R}^{n}$ is the input torques (forces) vector.

The disturbance term $\Gamma_d(t)$ accounts for possible external disturbances that are not accounted for in the dynamics. Furthermore, $\Gamma_d(t)$ may include the nonlinearities that emerge from the simplifying hypotheses. The formulation of the dynamics in (2.2) is the standard formulation widely used in robotics textbooks [Siciliano et al., 2009]. However, for compactness, all non-inertial terms are sometimes gathered in one single term. In this case, the dynamics of the manipulator (2.2) are rewritten as follows

$$M(q)\ddot{q} + N(q,\dot{q}) = \Gamma(t) \tag{2.3}$$

with $N(q,\dot{q}) \in \mathbb{R}^{n}$ gathers non-inertial terms, independent from the acceleration of the robot (Coriolis, gravity, etc.).

2.2.1 Linear in the parameters reformulation of the dynamic model

Model-based adaptive controllers rely mainly on a fundamental property of the manipulator's dynamics. This property consists in the linearity of the dynamics with respect to the dynamic parameters characterizing the manipulator mechanical structure such as inertia and masses [Khalil and Dombre, 2004; Craig et al., 1987; Siciliano et al., 2009]. Hence, the inverse dynamics could be rewritten in a linear form with respect to a chosen vector of parameters. This latter can include separate parameters or combinations of several ones (a product of the masses of two successive links for instance). Although this property might seem too restrictive, it is actually common to all rigid manipulators.

Consider the general form of the inverse dynamics of a m-DOF parallel manipulator given by (2.2). The external disturbances $\Gamma_d(t)$ do not usually appear in a linear form of the parameters. Hence, they are excluded from the reformulation of the dynamics. Consequently, the linear in the parameters reformulation of the dynamics can be expressed as:

$$Y(q,\dot{q},\ddot{q})\Phi + \Gamma_d(t) = \Gamma(t) \tag{2.4}$$

where $Y(.) \in \mathbb{R}^{n\times p}$ is called the regression matrix, which is a nonlinear function of $q, \dot{q}$ and $\ddot{q}$, and $\Phi \in \mathbb{R}^{p}$ is the vector of the parameters of the robot to be estimated.

It is worth noting that not all the dynamic parameters of the manipulators should be considered unknown/uncertain. Indeed, we may have a good knowledge about some parameters while other ones can be unknown, uncertain or time-varying. In this case, the set of parameters can be divided into two sets, namely, known and unknown parameters [Ortega and Spong, 1989]. A typical example is when considering the variations of the dynamic parameters of the robot's traveling plate due to payload handling. In such situation, it would be interesting to consider only the mass and inertia of the traveling plate to be estimated. In this case, the reformulation of the dynamics can be rewritten as follows:

$$Y_n(q,\dot{q},\ddot{q})\Phi_n + Y_u(q,\dot{q},\ddot{q})\Phi_u + \Gamma_d(t) = \Gamma(t) \tag{2.5}$$

where $\Phi_n \in \mathbb{R}^{p_1}$ contains known parameters and $\Phi_u \in \mathbb{R}^{p_2}$ contains only those parameters needed to be estimated [Ortega and Spong, 1989]. $Y_n(q,\dot{q},\ddot{q}) \in \mathbb{R}^{n\times p_1}$ and $Y_u(q,\dot{q},\ddot{q}) \in \mathbb{R}^{n\times p_2}$ are partial regression matrices.

2.3 Classification of existing control strategies

In view of the important impact of the control design on the overall performance of parallel manipulators, several of researchers investigated existing strategies and proposed novel ones in the quest of the perfect controller [He et al., 2007; Cheng et al., 2003]. Consequently, an abundance of control schemes in the literature for parallel manipulators can be noticed, varying from the simplest to the most complex ones.

The first control algorithms applied to PKMs were based on the classical linear PID controllers [Ziegler and Nichols, 1942]. However, it was observed that their performance was very limited, especially for high-speed tasks. Such a result is actually obvious since linear controllers do not take into account the nonlinear nature of the complex dynamics of parallel manipulators. It was then natural to propose more suitable and relatively complex control schemes that take the nonlinear dynamics into account. Based on this idea, nonlinear model-based controllers were then proposed, and widely adopted in order to improve the overall performance of PKMs [Cheng et al., 2003; Patel and George, 2012]. However, model-based schemes rely essentially on the dynamic model of the manipulator to be able to cancel the inherent nonlinearities in the dynamics. Consequently, if the modelled dynamics does not accurately match the real behavior of the system, the nonlinearities cannot be adequately compensated. In this case, such controllers could lead to bad performance or even instability of the closed-loop system.

Adaptive controllers were proposed as a more convenient way to deal with uncertainties and variations of the manipulator's dynamics [Craig et al., 1987; Nguyen et al., 1993]. Indeed, adaptive control schemes have a built-in capability of estimating and adjusting of

Figure 2.2 – Classification of existing control schemes for parallel manipulators

the controller's parameters to maintain the desired performance despite the uncertainties and variations of the dynamics of the PKM and its environment.

In the following section, the most relevant control methods, proposed in the literature, for PKMs are highlighted and explained. They can be divided into two categories; adaptive and non-adaptive controllers. The proposed classification is summarized in Figure 2.2.

2.4 Non-adaptive control schemes

Non-adaptive controllers are those which do not require any real-time adjustment of their parameters. Once the controller has been defined and its parameters (or feedback gains) have been tuned, it does not require any adjustment while operating. These control schemes can be divided into two main categories. The first category includes controllers that do not require any knowledge about the dynamics of the robot, while the second one consists of controllers requiring a dynamic model (model-based).

In the sequel, a quick overview on some of the most important control schemes belonging to the non-adaptive class is provided.

2.4.1 Non-model-based schemes

Non-adaptive non-model-based (also known as kinematic) controllers are the simplest control schemes that exist for parallel manipulators. These controllers do not require any knowledge about the dynamics of the manipulator. In most cases, the only required information is the state of the system (position and velocity). In practice, only a subset of the state vector is required, since the remaining elements could be estimated from the available information [Sartori Natal et al., 2014]. However, it should be noted that a good estimation is required to obtain a good performance.

PID control

The PID controller [Ziegler and Nichols, 1942] is the most used control strategy in industry. In fact, according to [Kiam et al., 2005], more than 90% of industrial controllers are still based upon PID algorithms. The popularity of the PID controller is mainly due to its simplicity and easy implementation compared to complex advanced controllers. This model independent control strategy provides a fairly acceptable performance with most real-world control problems.

A joint-space PID controller is the simplest controller that can be designed for PKMs. Its control law is given by:

$$\Gamma(t) = K_P e(t) + K_I \int_{t_0}^{t} e(\sigma) d\sigma + K_D \dot{e}(t) \tag{2.6}$$

where

$e(t) \triangleq q_d(t) - q(t)$ is the joint tracking error between the desired trajectory $q_d(t) \in \mathbb{R}^n$ and the measured one $q(t) \in \mathbb{R}^n$.

$K_P, K_I, K_D \in \mathbb{R}^{n\times n}$ are positive definite matrices representing the feedback gains.

If the design matrices are taken diagonal (i.e. no coupling between the joints is considered), the controller is called a linear single-axis controller.

In most practical cases, the only available information about the configuration of the robot is provided by the encoders located at the robot's joints. Consequently, a joint-space controller is the most suitable controller for such situations. Nevertheless, for the particular case where the pose of the end-effector could be directly measured, a Cartesian-space controller is often recommended. The control effort in this case is a force $F(t) \in \mathbb{R}^m$ to be applied to the traveling plate of the PKM making it track the desired trajectory. This control force can be expressed as follows:

$$F(t) = K_P e_c(t) + K_I \int_{t_0}^{t} e_c(\sigma)d\sigma + K_D \dot{e}_c(t) \tag{2.7}$$

where

$e_c(t) = X_d(t) - X(t)$ is the Cartesian trajectory tracking error of the-end effector, being $X_d(t) \in \mathbb{R}^m$ the desired trajectory and $X(t) \in \mathbb{R}^m$ the measured one.

$K_P, K_I, K_D \in \mathbb{R}^{m\times m}$ are positive definite feedback gains.

The actual control law to be applied to the actuators can be obtained by using the inverse Jacobian matrix $J_m \in \mathbb{R}^{n\times m}$. This latter is obtained through analyzing the kinematics of the PKM [Corbel et al., 2010b]. The inverse Jacobian matrix J_m maps the traveling plate velocity vector $\dot{X}$ to the joint velocity vector $\dot{q}$ according to the following equation:

$$\dot{q}(t) = J_m \dot{X}(t) \tag{2.8}$$

The inverse Jacobian matrix is also used to map the traveling plate's force vector F to the actuators' torques Γ according to the following equation:

$$F(t) = J_m^T \Gamma(t). \tag{2.9}$$

Hence, if the robot is non-redundant, the actuators' torques can be obtained by using the inverse of J_m. Otherwise, if the robot is redundant, a commonly used solution is to use the pseudoinverse of J_m, denoted by J_m^+ such that

$$\Gamma(t) = (J_m^+)^T F(t) \tag{2.10}$$

where $J_m^+ \triangleq (J_m^T J_m)^{-1} J_m^T$ is the pseudoinverse of J_m.

As previously mentioned, PID controllers provide a fair performance with most of real world systems. However, for parallel manipulators operating at high accelerations, a special care has to be considered. Such systems are known for their varying nature, coupled dynamics and nonlinear nature, which may result in an unstable closed-loop system. Moreover, PID controllers suffer from the lack of robustness and the need of retuning the gains if the experimental conditions change (e.g. additional payload).

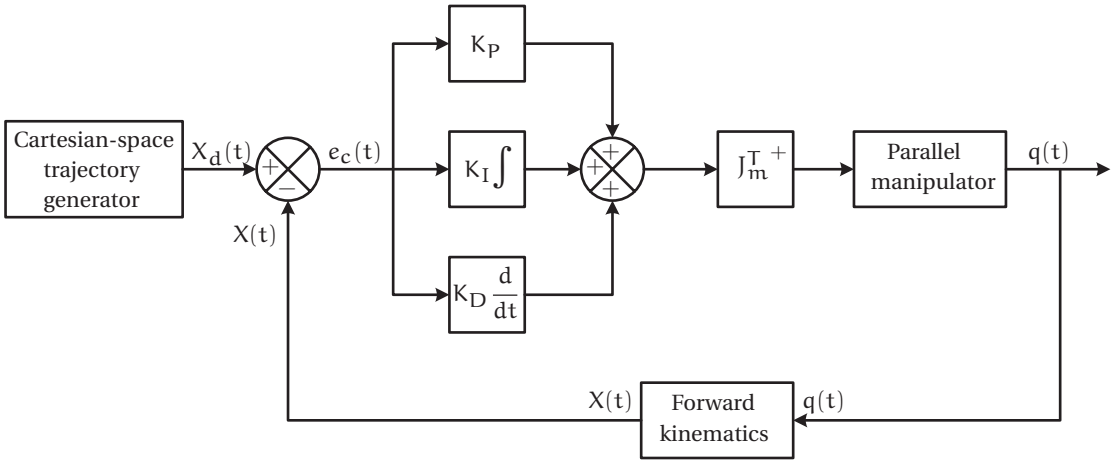


Figure 2.3 – Block diagram of Cartesian-space PID control for RA-PKMs

Nonlinear PID control

Nonlinear PID (N-PID) controllers are a modified version of the previously discussed, more common, linear PID controllers. They are often being promoted as an enhanced version of standard linear PID controllers. In fact, nonlinear PID controllers share a similar structure with their linear counterparts. However, instead of using constant feedback gains K_P, K_I and K_D, they are replaced by nonlinear time-varying ones that might depend on the system state, input or other variables [Shang et al., 2009]. Compared to linear ones, nonlinear PID controllers offer several advantages. Such improvements include guaranteed stability, better tracking performance and better robustness towards uncertainties and time-delays.

A general form of joint space N-PID control law can be expressed as follows

$$\Gamma(t) = K_P(.)e(t) + K_I(.)\int_{t_0}^{t} e(\sigma)d\sigma + K_d(.)\dot{e}(t) \tag{2.11}$$

where $K_P(.), K_I(.), K_D(.) \in \mathbb{R}^{n\times n}$ are nonlinear feedback gain matrices. The choice of the nonlinear gains is not unique and depends on several parameters such as the system dynamics, environment conditions, etc.

Similar to the linear PID controller, N-PID controllers can be applied in both joint and task spaces by simply replacing the constant feedback gains by nonlinear varying ones.

In [Su et al., 2004], a nonlinear PID controller in conjunction with a nonlinear tracking differentiator have been proposed for the control of a 6-DOF parallel manipulator in joint space. First, the usual derivative signal in conventional feedback controllers is replaced by the signal generated by the proposed differentiator. Then, the N-PID controller is designed based on nonlinear functions of the tracking error and some design parameters. Real-time

experiments revealed accurate tracking of the desired trajectories and robustness towards measurement noise thanks to the proposed tracking differentiator.

Fuzzy control

Fuzzy control can be thought of as a mathematical formulation of vagueness in linguistics. It can be seen as a generalization of the classical set theory with the peculiarity of considering logical variables taking continuous values between 0 and 1.

A Fuzzy logic based control strategy has been proposed for position control of a 6-DOF Stewart platform in [I-Fang et al., 1999]. The proposed fuzzy control scheme is based on IF-THEN rules describing the relation between the input variables (i.e. the joint tracking error and its rate) and the control output. A major contribution of this work is the stability analysis of the closed-loop system using the Popov criterion stability theorem. Real-time experiments were conducted on a Stewart platform with hydraulic actuators.

In [Begon et al., 1995], a fuzzy sliding mode control is proposed for the control of a 6-DOF PKM. Similar to [I-Fang et al., 1999], IF-THEN rules are used in this work to determine the control output magnitude based on the sliding variable and its first time-derivative. The provided simulation and experimental results illustrated a net superiority of the proposed controller compared to a standard PI controller.

It is worth mentioning that other works based on fuzzy logic exist in the literature relying on the dynamics of the PKM in the control loop. Since the current section is dedicated to non-model-based controllers only, such strategies cannot be included herein.

2.4.2 Model-based strategies

Strategies that belong to this category are relatively more complex than those of the previous one. Indeed, model-based controllers include the dynamic model of the manipulator in the control loop to enhance the performance of the system. Some strategies depend on part of the dynamics while more sophisticated ones require the whole dynamics to be modeled. Including the dynamics of the manipulator in the control loop reduces the effect of the nonlinearities of the system and improve the performance of the controlled system.

PD control with gravity compensation

The PD with gravity compensation controller is probably the simplest non-adaptive model-based controller. The main idea behind this controller is to include the gravity-related part of the dynamics in the control loop. Hence, the prior knowledge of part of the dynamics of the manipulator is required. The control law of the PD with gravity compensation controller is composed of a linear PD feedback term in addition to the gravity term.

This control law can be expressed in joint space as follows [Chifu et al., 2010]:

$$\Gamma = K_P e(t) + K_D \dot{e}(t) + G(q) \tag{2.12}$$

where $K_P, K_D \in \mathbb{R}^{n\times n}$ are positive definite feedback matrices and $e(t) \in \mathbb{R}^n$ is the joint tracking error. From a practical point of view, the PD with gravity compensation yields better tracking performance than a simple linear PID controller. Most importantly, from a theoretical point of view, the controller in (2.12) is able to satisfy the position tracking objective globally for a suitable choice of the gain matrices K_P and K_D.

Another variant of the controller in (2.12) is the so-called PD with desired gravity compensation [Kelly, 1997]. Instead of using the measured joint positions $q(t)$ when evaluating $G(q)$, which may be time consuming, we use the desired trajectories $q_d(t)$ instead. Then, if the desired trajectories are already preplanned, the desired gravity vector $G(q_d)$ can be computed offline, which reduces the computation time of the controller. However, to the best of our knowledge, the PD controller with desired gravity compensation has never been applied to PKMs.

It is worth noting that compensating the gravity in the case of PKMs operating at high accelerations does not have significant outcomes on the performance of the system. This is mainly due to the fact that the effect of the gravitational forces is negligible compared to the other effect (inertia and Coriolis).

Inverse dynamics-based strategies

PD control plus feedforward. The PD with computed feedforward is one of the most efficient non-adaptive model-based controllers that exist for parallel manipulators. The main idea behind this controller is to use the full inverse dynamic model of the manipulator in the control loop in order to enhance the overall performance [Sartori Natal et al., 2012]. Many limitations however arise when evaluating the inverse dynamic model. Indeed, this latter is complex and highly nonlinear, hence, it will be time consuming to be evaluated it in real-time. Moreover, the measured quantities are often noisy, which may deteriorate the performance of the closed-loop system. Furthermore, velocities and accelerations are usually non-measurable, since most of real world manipulators are only equipped with position sensors (i.e. encoders).

The PD with computed feedforward overcomes all the aforementioned limitations by using the desired trajectories $q_d(t), \dot{q}_d(t), \ddot{q}_d(t) \in \mathbb{R}^n$ instead of $q(t), \dot{q}(t), \ddot{q}(t) \in \mathbb{R}^n$. The joint space control law of this controller is then given by

$$\Gamma(t) = K_P e(t) + K_D \dot{e}(t) + M(q_d)\ddot{q}_d + C(q_d, \dot{q}_d) + G(q_d) \tag{2.13}$$

where $K_P, K_D \in \mathbb{R}^{n\times n}$ are positive definite feedback matrices and $e(t), \dot{e}(t) \in \mathbb{R}^n$ are the joint position and veocity tracking errors, respectively. As it can be noticed, the control law

in (2.13) is based on the robot dynamics evaluated being using the desired positions, velocities and accelerations. Consequently, the feedforward dynamics can be computed offline and stored in memory, which is practical for setups with modest hardware configuration.

Augmented PD (PD+) control. The augmented PD (also known as PD+) controller is one of the simplest control schemes that guarantee the control objective globally, provided that the centrifugal and Coriolis term is determined using Christoffel symbols (i.e. can be written in the form $C(q,\dot{q})\dot{q}$). The control law of the PD+ control law is quite similar to that of the PD with feedforward, except that, instead of only using the desired trajectories, the dynamics are evaluated using both measured and desired trajectories. The PD+ control law is given by [Zhang et al., 2007]:

$$\Gamma(t) = K_P e(t) + K_d \dot{e}(t) + M(q)\ddot{q}_d + C(q,\dot{q})\dot{q}_d + G(q) \tag{2.14}$$

where $K_P, K_D \in \mathbb{R}^{n\times n}$ are positive definite feedback matrices and $e(t), \dot{e}(t) \in \mathbb{R}^n$ are the joint position and velocity tracking errors, respectively. Similar to other non-adaptive model-based strategies, the PD+ requires accurate knowledge of the dynamic model of the manipulator in order to reach the desired performance.

Computed torque control. As it was described earlier, the dynamics of PKMs in (2.2) is known for its nonlinear nature. The fact that model-based controllers are also composed of nonlinear terms gives the impression that the resulting closed-loop system is nonlinear as well. This is indeed the case for all the previously described model-based controllers, but not for the computed torque controller.

The computed torque controller is a well-known strategy that comes with a unique feature. Compared to other non-adaptive model-based controllers, the computed torque control allows to obtain a linear closed-loop equation in terms of the tracking errors of the manipulator. The control law of the computed torque in joint-space is given by [Asgari and

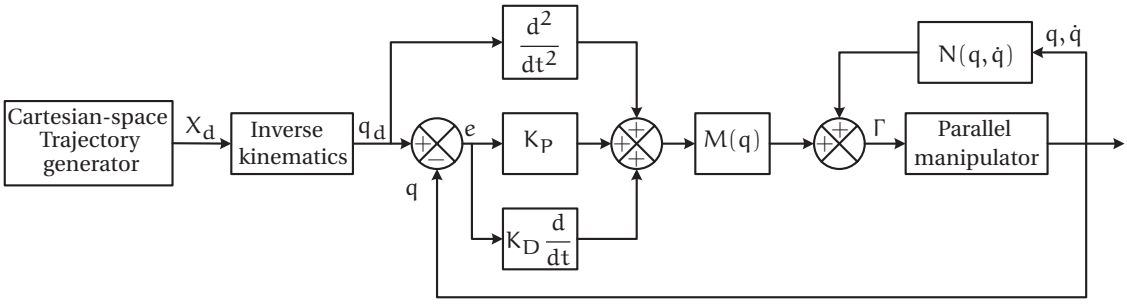


Figure 2.4 – Block diagram of computed torque control

Ardestani, 2015]:

$$\Gamma(t) = M(q)\left[\ddot{q}_d + K_P e(t) + K_D \dot{e}(t)\right] + C(q,\dot{q})\dot{q} + G(q) \tag{2.15}$$

where $K_P, K_D \in \mathbb{R}^{n\times n}$ are positive definite feedback matrices and $e(t), \dot{e}(t) \in \mathbb{R}^n$ are the joint position and veocity tracking errors, respectively. Substituting (2.15) into the dynamics of the manipulator in (2.2) (without considering friction and external disturbances) leads to the following linear error system:

$$\ddot{e}(t) + K_D \dot{e}(t) + K_P e(t) = 0 \tag{2.16}$$

As it can be noticed, the computed torque control law seeks for full compensation of the nonlinearities of the manipulator. To that aim, it is necessary to have exact knowledge of the system's parameters in real-time, which might be a tedious task since these parameters are most likely to vary during functioning. Moreover, it is necessary to have accurate measure of the joint positions and velocities, which is impossible in practice due to finite resolution of the encoders. Hence, practically speaking, the linear behavior of the tracking error described in (2.16) could never be exactly achieved.

A block diagram illustrating the joint space computed torque controller (2.15) is shown in Figure 2.4.

Strategies with nonlinear feedback gains

Model-based control schemes with nonlinear feedback gains are relatively new in robotics. These strategies are mainly based on conventional model-based strategies such as computed torque, augmented PD, etc. The main difference is the substitution of the linear feedback loop, commonly used in conventional controllers, by a nonlinear time-varying one. This proposition is motivated by the advantages that nonlinear gains hold compared to their linear counterparts. Hence, any model-based controller with constant feedback gains can be extended by including nonlinear time-varying ones.

[Shang et al., 2009] proposed to extend the standard Cartesian space augmented PD controller by using nonlinear feedback gains instead of the usual constant ones. The proposed control law in this work consists of three distinct terms; a nonlinear feedback term based on [Han, 1994], a dynamic compensation term and a friction compensation term. Stability proof of the proposed controller based on the *Lyapunov* theory and *Barbalat*'s lemma was provided. Real-time experiments were conducted on a 2-DOF redundantly actuated PKM. A comparative study with a conventional augmented PD controller revealed a significant tracking performance improvement with the proposed controller.

In [Shang et al., 2012], the Cartesian-space computed torque controller has been revisited with nonlinear feedback gains. The same procedure in [Shang et al., 2009] was followed to demonstrate the relevance of using time-varying nonlinear gains instead of

constant ones. Real-time experiments on a 2-DOF redundantly actuated parallel manipulator show a net superiority of the proposed controller with respect to the conventional one in terms of tracking performance.

Sliding mode control

Sliding Mode Control (SMC) is robust nonlinear control strategy that is capable of providing the desired performance of the control system despite the presence of disturbances and uncertainties [Perruquetti and Barbot, 2002]. The design of SMC schemes consists mainly in two fundamental steps; the first is to ensure the reachability phase and the second is to drive the states of the system to the equilibrium point. In the reachability phase, the controller drives the states to a stable domain called the sliding surface. Then, during the sliding phase, the controller keeps the states on the surface while sliding to the equilibrium point.

Thanks to their robustness features, various SMC and SMC-based controllers were proposed in the literature for parallel manipulators. [Jafarinasab et al., 2011] proposed to apply SMC in Cartesian-space for the control of a 6-DOF parallel manipulator. To show the capabilities of the SMC, two simulation scenarios were conducted on a parallel manipulator for a circular path tracking of its moving platform. In the first scenario, the exact parameters of the manipulator were used in the control law. The second scenario considered large uncertainties in the mass properties of the moving platform. It was concluded that SMC provides fast transient response and accurate trajectory tracking.

In [Qi et al., 2007], a fuzzy sliding mode controller was proposed for the Cartesian-space trajectory tracking of a 4-DOF parallel manipulator. In the first step, the considered control law was based on to the one in [Jafarinasab et al., 2011] which is a basic SMC. However, to avoid chattering problems that occur due to the use of discontinuous control law, fuzzy logic systems were proposed to replace the discontinuous control term. Numerical simulations were performed to demonstrate the effectiveness of the proposed controller.

Predictive control

Predictive control is an advanced control technique typically used for industrial systems. The main idea behind predictive control is to predict the behavior of the system based on its dynamic model. The control law is then formulated as the solution of an optimization problem with constraints based on the future behavior of the system [Maciejowski, 2002]. Predictive control strategies involve receding horizon optimization that might be time consuming. Consequently, such strategies are typically used for slow systems such as chemical plants and oil refineries. For this reason, predictive control is uncommon in PKMs, which are extremely fast systems that require very small sampling rates.

[Vivas et al., 2003] proposed to apply predictive control to the H4 PKM [Pierrot et al., 2001]. First, the nonlinear dynamics of the manipulator were linearized and decoupled

using nonlinear feedback techniques. Then, predictive control is formulated based on the linearized decoupled dynamics. The proposed is experimentally validated on the H4 robot and provided satisfactory tracking performance compared to computed torque control.

The block diagram summarizing the proposed controller in [Vivas et al., 2003] is shown in Figure 2.5.

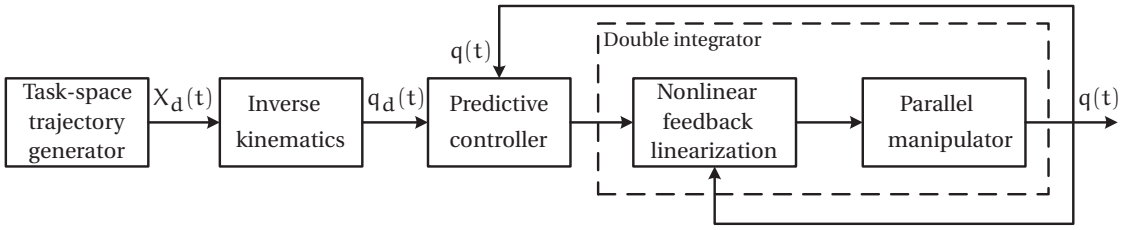


Figure 2.5 – Bloc diagram of Predictive Control for parallel manipulators

Robust H_∞ control

H_∞ control theory deals with the problem of synthesizing control laws that achieve stability and guarantee performance despite the presence of disturbances acting on the system [Zhou and Doyle, 1998].

A typical example of application of H_∞ control theory on PKMs can be found in [Becerra-Vargas and Belo, 2012]. In this work, the computed torque controller is first applied to the PKM to obtain a linear error dynamics. Then, the closed-loop system is augmented by the H_∞ controller. The uncertainties resulting from the inaccuracies in the model parameters were considered as an external disturbance acting on the error dynamics. These uncertainties have to be accounted for by the robust H_∞ controller. The proposed controller was validated through numerical simulations on a 6-DOF Stewart-platform flight simulator. A comparative study showed a better noise effect rejection with the H_∞ controller compared with the standard non-augmented computed torque controller.

Another application of H_∞ control to PKMs can be found in [Rachedi et al., 2015]. In this work, a tangent linearization procedure of the nonlinear dynamics of the PKM is used to obtain a multivariable linear model around a functional point. Then, the H_∞ controller is developed based on the obtained linearized model by the mixed sensitivity approach [Zhou and Doyle, 1998]. The proposed controller was applied to the 3-DOF Delta robot and showed improved performance at high accelerations (12g) compared with a standard PID controller. Furthermore, experimental results of payload change scenario revealed a better robustness of the H_∞ controller compared with PID.

2.5 Adaptive control strategies

Controllers that belong to the class of adaptive schemes require a real-time update of their parameters in the aim of finding their ideal values. The development of adaptive strategies is motivated by the abundance of uncertainties in the controlled system and its environment that may deteriorate the control performance. Analogously to non-adaptive controllers, this class can be also divided into non-model-based and model-based strategies.

In what follows, we give an overview on the most prominent adaptive control schemes that can be found in the literature.

2.5.1 Non-model-based adaptive strategies

Strategies belonging to this class do not require the dynamics of the system nor its structure to be known. Instead, they consider all the nonlinearities of the system due to uncertainties and possible disturbances, whose structure is unknown, as a general disturbance term to be estimated in real-time. Then, the control law is designed such that the effect of this disturbance is minimized.

MRAC-based startegies

The main idea of Model Reference Adaptive Control (also called MRAS for Model Reference Adaptive System) is to obtain a closed-loop system with adjustable controller parameters. These parameters have the ability of changing the behavior of the closed-loop system. The time evolution of the adaptive parameters is adjusted by comparing the output of the controlled system and the desired one obtained from the reference model. The ultimate goal is to make the behavior of the controlled system match a desired reference model despite the eventual variations/uncertainties in the system or its environment [Astrom, 1995].

While numerous MRAC-based control schemes for serial manipulators can be found in the literature [Dubowsky and DesForges, 1979; Nicosia and Tomei, 1984], only few controllers were proposed for parallel ones. In [Nguyen et al., 1993], an adaptive control scheme based on MRAC has been proposed to control a 6-DOF parallel manipulator based on the Stewart platform prototype used to emulate space operations. In this work, the control scheme consisted of a joint space PD feedback controller with adjustable gains. The control scheme is further endowed with an adaptation mechanism that controls the evolution of the adaptive feedback gains. The joint tracking error of each axis was supposed to follow a desired user-defined reference second order linear system characterized by its natural frequency and its damping ratio. Based on *Lyapunov* stability analysis, the adaptation law stabilizing the error dynamics is derived. Experiments were conducted on a Stewart-platform PKM to evaluate the performance of the controller and its robustness towards

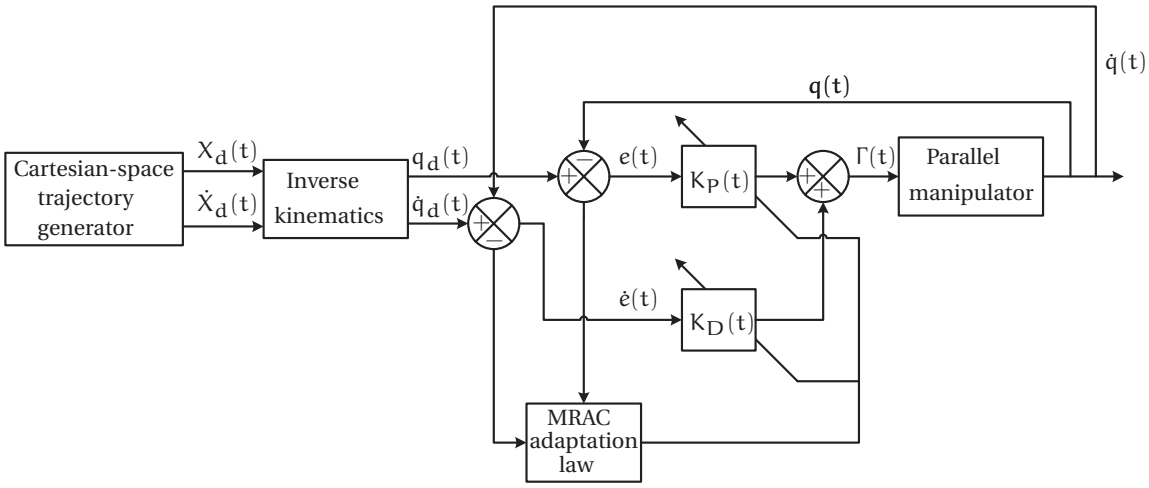


Figure 2.6 – Bloc diagram of the proposed MRAC-based control in [Nguyen et al., 1993]

sudden payload changes. Two case studies were considered; mainly, a vertical motion of the moving platform and a circular one. In both cases the proposed MRAC-based strategy with adaptive gains outperforms the PD with constant feedback gains. Figure 4.1 illustrates the block diagram summarizing the MRAC-based controller proposed in [Nguyen et al., 1993].

Strategies based on artificial neural networks

Artificial Neural Networks (ANN) are known by their powerful universal approximation features [Yegnanarayana, 2009]. This is why they attracted a big deal of interest in various fields, and have been applied almost everywhere, from identification to estimation and control. Artificial neural networks learn from experience rather than programming, hence, they are typically used in repetitive tasks. Several works can be found in the literature regarding the application of neural networks in control of PKMs [Qingsong and Yangmin, 2007]. However they remain relatively few compared to what can be found for serial manipulators [Lewis et al., 1998].

Thanks to their learning ability, artificial neural networks are mostly used to approximate the dynamics of the manipulator. Then, the learned dynamics can be included in the control scheme to compensate for the uncertainties and disturbances. [Li and Wang, 2009] proposed to augment a decentralized Cartesian space PID controller with an artificial neural network term. The main motivation of this work is to improve the tracking capabilities of a 2-DOF redundantly actuated parallel manipulator. The provided simulation results demonstrated the superiority of the augmented controller with respect to the original PID.

The maximum errors were significantly reduced since the additional neural network term accounted for the nonlinear dynamics of the manipulator.

2.5.2 Model-based adaptive strategies

Model-based adaptive controllers explicitly include the dynamics of the manipulator in the control loop. In contrast to their non-adaptive counterparts, the adaptive schemes adjust the parameters of the model-based loop in order to converge them their best steady-space values. Consequently, if the dynamics of the manipulator is uncertain or time-varying, adaptive controllers result in a better closed-loop performance.

Strategies based on computed torque

Computed-torque-based adaptive schemes are those controllers relying on the classical non-adaptive computed torque controller. The control design starts by considering the ideal non-adaptive controller that assumes perfect cancellation of the system nonlinearities. Due to possible uncertainties and variations in the system dynamics, the compensation of the nonlinearities of the system could not be as perfect as the non-adaptive controller assumes. Then, an additional adaptive estimation loop is considered to account for these uncertainties. The role of the adaptive loop is to estimate, in real-time, the system's parameters and the obtained estimation are used in the controller.

The typical example of a computed-torque-based adaptive controller is the one proposed in [Craig et al., 1987]. The first step of the control design in this work is to consider a computed torque control law with uncertain parameters. Then, the error equation resulting from the application of this control law is obtained. Based on a thorough stability analysis using the *Lyapunov* theory, an adaptive law for the parameters' estimation is derived. The proposed adaptive controller in conjunction with the adaptation law guarantee that the tracking errors vanish and that the estimated parameters converge to their best

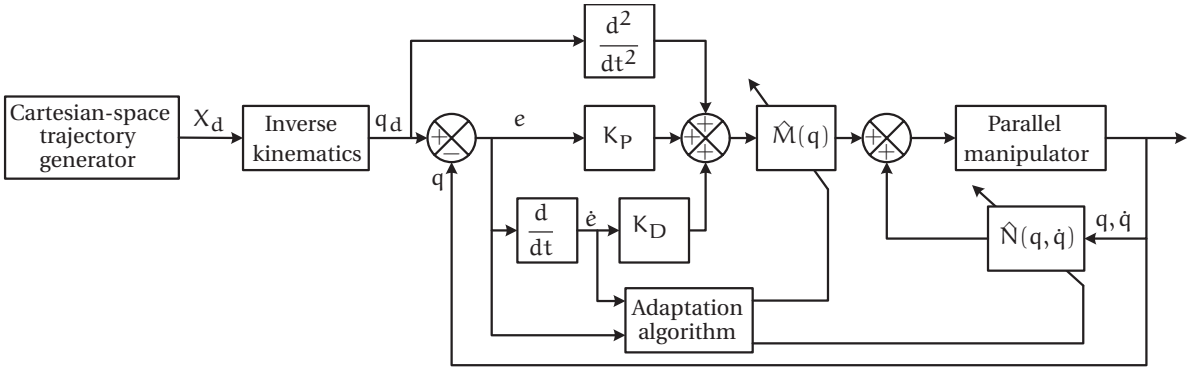


Figure 2.7 – Bloc diagram of adaptive computed torque control

steady-state values. The block diagram depicted in Figure 2.7 clarifies the principle of such control strategy.

The computed-torque-based adaptive controller, proposed in [Craig et al., 1987], was mainly developed for serial manipulators. However, it is known that parallel manipulators share many properties with their serial counterparts [Cheng et al., 2003; Merlet, 2006]. Based on this fact, the controller in [Craig et al., 1987] has been straightforwardly applied to PKMs in [Shang et al., 2012]. In this work, an adaptive Cartesian space computed-torque-based scheme is applied to a 2-DOF redundantly actuated PKM. Real-time experiments were carried out in order to highlight the benefits of the adaptive controller compared to its non-adaptive version. The adaptive controller shows a net superiority and allows to estimate the system dynamic parameters.

A main drawback of computed torque based adaptive controllers is their dependence on the real acceleration of the robot [Craig et al., 1987]. This shortcoming is crucial from a practical point of view since measuring actual accelerations is tedious.

Based on passivity of the system

Passivity-based controllers use the passivity property of the manipulator [Ortega and Spong, 1989]. In contrast with computed-torque-based ones, passivity-based controllers do not assume perfect cancellation of the system nonlinearities, even in the ideal case of perfect knowledge of the manipulator's parameters. Hence, they do not lead to a closed-loop linear error system. However, based on the passivity property of the system, they result in a stable closed-loop system.

[Sadegh and Horowitz, 1990] proposed a passivity-based adaptive controller that holds many advantageous features. This controller can be assimilated to an adaptive version of PD control with computed feedforward. Indeed, the proposed control law consists mainly in a PD feedback loop in addition to an adaptive feedforward term based on the dynamics of the robot and the desired reference trajectories. The main advantage of this scheme is that it does not require the joint acceleration measurement. Moreover, the compensation of the system's nonlinearities is based on desired quantities that can be computed offline. This means that the control scheme does not require heavy computations. Furthermore, the use of the desired values instead of the measured ones, which can be often noisy, may enhance the robustness of the controller toward measurement noise.

In [Honegger et al., 2000], the developed controller in [Sadegh and Horowitz, 1990] has been embraced for joint space control of a 6-DOF PKM called the Hexaglide. The proposed adaptive controller as well as a linear PD controller were implemented in real-time for a comparison purpose. It was not surprising that the adaptive controller provided better tracking results since it compensates for the nonlinear dynamics of the manipulator.

In [Sartori Natal et al., 2015], a Cartesian space control strategy called *Dual-Space Adaptive Control* is proposed to control RA-PKMs. The proposed controller in this work share many similarities with the proposed control strategy in [Honegger et al., 2000]. The main

difference is that the weighting gain of the position error with respect to the velocity error is time-varying. The proposed controller was experimentally validated on the R4 parallel manipulator [Corbel et al., 2010a]. Real-time experiments showed that the proposed Dual-Space adaptive controller significantly improves the tracking performance and allows the tracking of extremely fast trajectories (up to 100G of maximum acceleration).

[Shang and Cong, 2010] proposed to extend the joint-space passivity based adaptive controller of [Slotine and Weiping, 1987], mainly developed for serial manipulators, to Cartesian-space control of parallel manipulators with actuation redundancy. Moreover, to deal with friction an additional control term has been added to the conventional feedback and adaptive loops. Real-time experiments, conducted on a 2-DOF RA-PKM, showed the superiority of the proposed adaptive controller compared to the augmented PD controller in terms of Cartesian tracking errors, at both low and high accelerations. Unfortunately, the evolution of the estimated parameters was not shown in this work to see if they really converge to their steady-state values.

2.6 Conclusion

This chapter presented an overview of the main control schemes proposed in the literature for parallel manipulators. The existing controllers were divided into two main categories; non-adaptive and adaptive classes. Then, each category is divided into two classes based on their dependence or not on the dynamic model of the manipulator.

Adaptive controllers provide better closed-loop performance in the presence of uncertainties in the manipulator's dynamics as well as its environment. However, they may involve a large set of design parameters that require a careful tuning. Non-adaptive controllers are relatively easy and intuitive to adjust. Nevertheless, in the presence of uncertainties, these control strategies may not lead to a good performance.

Model-based controllers involve the use of the dynamic model of the parallel manipulator in the control loop. Though these strategies result in a superior performance, their practical implementation may be challenging. From one hand, they can be very cumbersome due to the complex nonlinear nature of the dynamics to be computed in real-time. From the other hand, some strategies may require non-measurable signals of the robot to be available at each sample time (e.g. real accelerations).

On the basis of the above discussion and on what exists in the literature of parallel manipulators, it is now clear that adaptive controllers are the most appropriate to satisfy our needs. Consequently, we decide to embrace the theory of adaptive control in the aim of finding the best controllers to reach the desired performances and specifications of the ARROW project. Specifically, in terms of accurate and high-speed trajectory tracking, in both joint and Cartesian spaces.

Part II

Proposed control solutions

CHAPTER

3 Solution 1: Enhanced model-based adaptive control

Contents

3.1 Introduction

In this chapter, two novel model-based adaptive control schemes are proposed for the control of parallel manipulators. The first one is inspired from a passivity-based adaptive control scheme called ***D**esired **C**ompensation **A**daptation **L**aw* (**DCAL**) [Sadegh and Horowitz, 1990]. This controller has been selected because of its various advantages compared to other existing approaches. Mainly, its simple architecture, its robustness towards measurement noise and its modularity. The second proposed controller is based on a recently developed robust nonlinear feedback control strategy called ***R**obust **I**ntegral of the **S**ign of the **E**rror* (**RISE**) [Xian et al., 2004]. This latter is a non-model-based control scheme which, as its name indicates, features a signum term that accommodates for a large class of uncertainties in the system. A theoretical background on both original DCAL and RISE will be provided first, before addressing the proposed extensions, improvements and their application to PKMs.

3.2 Contribution 1: DCAL with nonlinear feedback gains

3.2.1 Background on DCAL

DCAL is an adaptive control strategy proposed for the first time in [Sadegh and Horowitz, 1990] for the control of serial manipulators. It belongs to the family of passivity-based control strategies since it relies on the passivity nature of the manipulator. DCAL was inspired from an exact compensation control scheme consisting of a PD feedback-loop plus a model-based adaptive term. This latter is based on the dynamic model of the manipulator being evaluated using the measured state (i.e. position and velocity). Moreover, the adaptation control law is also based on the measured states which may be noisy. The main motivation behind the development of DCAL is to replace the measured states in the control law of the exact compensation controller by the desired ones, while still satisfying the control objective of the original controller.

Control law

Before proceeding to the development of DCAL control law, let us briefly recall the equations of dynamics of PKMs. Consider the dynamics of a m-DOF PKM with n actuators (such that $n \geq m$) given in joint space form as follows [Siciliano et al., 2009]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + \Gamma_d(t) = \Gamma(t) \tag{3.1}$$

where $M(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is the Coriolis and centrifugal forces matrix and $G(q) \in \mathbb{R}^n$ is the gravity forces vector. $q, \dot{q}, \ddot{q} \in \mathbb{R}^n$ are the joint position, velocity and acceleration vectors, respectively. $\Gamma \in \mathbb{R}^n$ is the control input vector and $\Gamma_d \in \mathbb{R}^n$ gathers all disturbances acting on the PKM.

Since DCAL belongs to model-based adaptive control strategies, it is necessary to rewrite the dynamics in (3.1) in a linear in the parameters form as follows [Craig et al., 1987]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = Y(q,\dot{q},\ddot{q})\Phi \tag{3.2}$$

where $Y(.) \in \mathbb{R}^{n\times p}$ is the regression matrix, function of joint positions, velocities and accelerations and $\Phi \in \mathbb{R}^n$ is a vector of constant parameters.

DCAL control law consists mainly of an inverse dynamics feedforward term being evaluated using the desired positions, velocities and accelerations and a linear PD feedback-loop. In order to accommodate for the errors emerging from replacing the actual measured states by desired ones, an additional nonlinear feedback function is appended to the control law. The joint space control law of DCAL in is expressed by [Sadegh and Horowitz, 1990]:

$$\Gamma_{DCAL}(t) = Y(q_d,\dot{q}_d,\ddot{q}_d)\hat{\Phi}(t) + K_P e(t) + K_V e_v(t) + \sigma \left\| e(t) \right\|^2 e_v(t) \tag{3.3}$$

where

$e(t) \triangleq q_d(t) - q(t)$ is the joint position tracking error

$e_v(t) \triangleq \dot{e}(t) + \lambda e(t)$ is the combined position-velocity tracking error.

K_P, $K_V \in \mathbb{R}^{n \times n}$ are constant linear feedback gain matrices, usually chosen diagonal.

The choice of a combined position-velocity tracking error e_v in the control law is inspired from sliding mode control techniques. This specific choice avoids that the tracking errors diverge at a constant rate and prevent the parameters drift. $\hat{\Phi}(t)$ is an on-line estimation of the parameters vector Φ, and σ is a positive gain.

Adaptation law

The evolution of the estimated parameters $\hat{\Phi}(t)$ in (3.3) is governed by the following adaptation law:

$$\dot{\hat{\Phi}}(t) = K Y^T(q_d, \dot{q}_d, \ddot{q}_d) e_v(t) \tag{3.4}$$

where $K \in \mathbb{R}^{p \times p}$ is a diagonal positive definite adaptation gain matrix. As it can be seen, the regression matrix $Y(.)$ in the adaptation law (3.4) is also evaluated using desired trajectories instead of the measured ones. The block diagram illustrating the DCAL control strategy is shown in Figure 3.1.

Advantages and limitations

As it was previously mentioned, DCAL was inspired by an exact compensation adaptive strategy that requires that the actual joint positions and velocities should be available. The main feature of DCAL compared to other adaptive strategies is the use of the desired quantities instead of the measured ones, both in the control and adaptation loops. This particular choice have great advantages, especially from a practical point of view. The main advantages of DCAL can be summarized in the two following points:

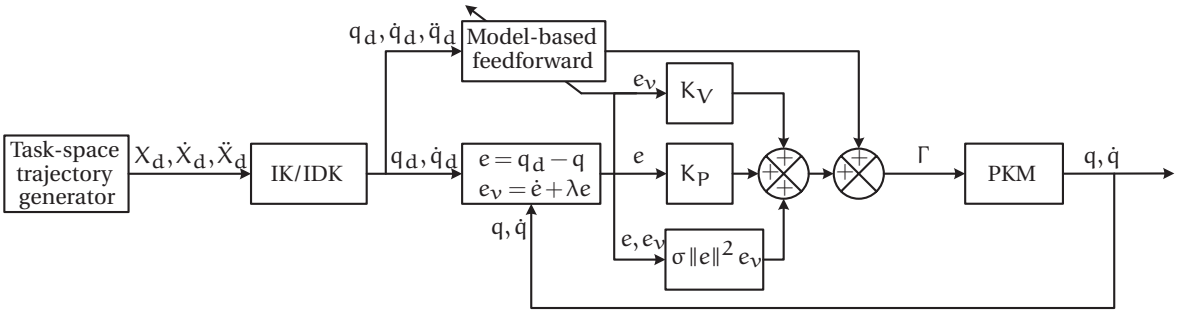


Figure 3.1 – Block diagram of DCAL control scheme, IK: Inverse Kinematics, IDK: Inverse Differential Kinematics.

- The amount of online computations of the controller is significantly reduced. Indeed, the feedforward dynamics can be computed offline and stored. Then, this term can be used in real-time, without the need to be computed online. This can be vital for setups with limited hardware, where the complex nonlinear dynamics cannot be evaluated in real-time.
- The robustness of the controller is enhanced since the control law and the adaptation algorithm heavily rely on the desired trajectories instead of measured ones which may be noisy and inaccurate. Usually, a filtering of the measurements is required to cope with the noise, which may induce a time-delay and make the controller more complex. Moreover, most real-world manipulators are not equipped with velocity sensors, thus joint velocities are in this case estimated from the measured position which may be inaccurate and noisy [Sartori Natal et al., 2014].

For these reasons, it can be concluded that desired compensation strategies in general, and DCAL in particular, are very adequate for real-time implementations. Nevertheless, DCAL control law can be enhanced if a careful attention to the PD feedback term is considered. Indeed, the DCAL control law features a linear PD feedback-loop with conventional constant gains. Even though this simple choice is enough to ensure the stability of the closed-loop system, the overall performance can be further enhanced if a careful attention is given to the PD feedback-loop. One possible solution to enhance the performance of DCAL would be to consider time-varying gains in the feedback loop instead of the conventional constant ones. Such a proposition would endow the closed-loop system with the qualities of nonlinear feedback control in addition to the robustness and adaptation features of DCAL.

3.2.2 From linear to nonlinear feedback gains

Basic idea

Conventional PID algorithms involve three constant gains being multiplied by the error, its integral and its derivative, respectively. Each control action (i.e. Proportional, Integral or Derivative) has a specific role in the controller in order to achieve a desired performance. Classical algorithms consider feedback gains that remain constant until manually changed.

Nonlinear feedback gains have the same working principle as the linear ones in the sense that they are multiplied by the different forms of the error. The main difference, however, is that the gains are no longer constants. Indeed, nonlinear gains are time-varying and depend on some information about the system. Such information can include the system state, the control input and other variables. An illustrative curve of the difference between linear and nonlinear feedback functions is depicted in Figure. 3.2.

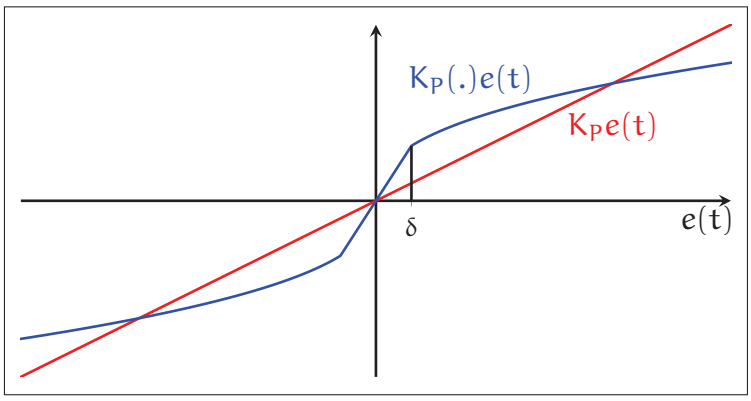


Figure 3.2 – Illustration of the evolution of a linear and a nonlinear feedback action versus the tracking error.

Some examples

One main advantage of nonlinear feedback controllers is that the choice of the nonlinear function is not unique. Indeed, in the literature, several nonlinear PD/PID controllers have been proposed. The control objective and stability analysis usually guide such a choice. In what follows, we enumerate some examples of N-PD/N-PID controllers proposed in the literature.

In [Xu et al., 1995], a N-PD controller has been proposed for the setpoint force control of a dexterous robotic hand. The proposed control law consists of a N-PD feedback loop in addition to a dynamics-based nonlinear term responsible for compensating the known nonlinearities. The proposed nonlinear proportional and derivative gains have the following common formula:

$$K(.) = \frac{K_{max}}{1 + \beta \exp(\alpha \, \mathrm{sgn}(\dot{e})e)} + K_{min} \tag{3.5}$$

where $K_{min}, K_{max}, \alpha, \beta$ are design parameters and e and $\dot{e}$ are position and velocity tracking errors, respectively. The choice of the nonlinear gains expressed by (3.5) enables having increasing gains when moving away from the setpoint and decreasing gains when moving toward the setpoint. Experimental results showed that the N-PD performed better compared to its linear counterpart, regarding both the tracking performance and energy consumption.

A joint space N-PD controller for mechanical manipulators has been proposed in [Kelly and Carelli, 1996]. The main contribution of the proposed controller is that global asymptotic stability is guaranteed while the control inputs remain inside some prescribed bounds. The nonlinear proportional and derivative gain matrices have been chosen diagonal, which means that no coupling between the joints is considered. The following

nonlinear functions have been proposed as a base for the nonlinear gains:

$$K_{Pi}(e_i) = \frac{k_{p_i} e_i}{a_i + |e_i|} \tag{3.6}$$

$$K_{Di}(\dot{q}_i) = \frac{k_{d_i} \dot{q}_i}{b_i + |\dot{q}_i|} \tag{3.7}$$

where i denotes the i^{th} joint, k_{p_i}, a_i, k_{d_i} and b_i are design parameters of the i^{th} joint's nonlinear gain, e_i and q_i are the position tracking error and the velocity of the i^{th} joint, respectively. The proposed controller has been experimentally implemented on a 2-DOF serial manipulator. The obtained results highlight the superiority of the N-PD controller compared to the classical PD control, especially regarding the tracking performance.

In [Gao et al., 2001], a general nonlinear control scheme has been presented for improved closed-loop performance. To illustrate the benefits of the proposed nonlinear gains, a comparison between a linear proportional controller and a nonlinear one has been conducted. It was shown that the steady-state error can be made arbitrary small in the case of the nonlinear controller while it remains relatively high in the case of the linear one. Based on the proposed nonlinear function, a generic PID controller has been developed. The control input of the proposed N-PID controller is expressed as follows:

$$u(t) = k_p f(e, \alpha_p, \delta_p) + k_i f(e_i, \alpha_i, \delta_i) + k_d f(\dot{e}, \alpha_d, \delta_d) \tag{3.8}$$

$$e_i(t) = \int_0^t e(\sigma) d\sigma \tag{3.9}$$

where $k_p, k_d, k_i, \alpha, \delta$, are positive design parameters. The nonlinear function $f(.)$ is defined as follows

$$f(x, \alpha, \delta) = \begin{cases} |x|^{\alpha} \operatorname{sgn}(x), & |x| > \delta, \\ x\, \delta^{\alpha-1}, & |x| \leq \delta, \end{cases} \qquad \delta > 0 \tag{3.10}$$

The choice of the nonlinear function (3.10) enables an interesting behavior of the feedback gains. Indeed, choosing $\alpha < 1$ for the proportional gain yields a small gain for large errors, which can help with windup problems for instance. On the other hand, choosing $\alpha > 1$ for the proportional gain yields a large gain for large errors allowing better tracking of the desired trajectory.

3.2.3 Proposed contribution: DCAL control with nonlinear feedback gains

As discussed previously, DCAL control scheme consists mainly of a fixed PD feedback loop, an adaptive feedforward term and a nonlinear compensation function. The stability analysis of DCAL demonstrated that it ensures the asymptotic tracking of reference trajectories as well as the boundedness of the estimated parameters [Sadegh and Horowitz,

1990]. Moreover, DCAL is known for its simplicity, efficiency and robustness thanks to its desired compensation design.

Stability analysis of DCAL states that the tracking errors $e(t)$ and $e_v(t)$ converge asymptotically to zero. However, it does not say much about the transient performance of the closed-loop system. In fact, during transient phase, the estimated parameters may be away from their best steady-state values or oscillating around them. This means that the inherent nonlinearities of the manipulator are not properly compensated. Consequently, the overall control scheme relies mainly on the feedback loop to achieve the control objective. The PD feedback loop will be responsible of both tracking the reference trajectories and rejecting the time-varying nonlinearities.

However, it is known in control theory that classical linear PD/PID controllers are not very suitable for control of nonlinear systems. Indeed, as explained earlier, linear feedback loops do not perform well with nonlinear systems and may lead to undesirable behavior. Consequently, the resulting closed-loop system may have a poor performance, especially in high accelerations. Moreover, linear PD/PID controllers are not robust enough to external disturbances and measurement noise. Consequently, it would be interesting to rethink DCAL with a carefully designed feedback loop that considers the nonlinear nature of the manipulator's dynamics. The proposed enhancement to DCAL should preserve all its advantages while improving the overall closed-loop behavior, with more focus on the tracking performance.

Proposed control law

In order to improve the tracking performance of DCAL, we would like to have the following properties. When the tracking error is large, the controller should supply high control effort for fast response time. From the other hand, the controller is supposed to provide low control effort on small tracking errors to avoid large overshoots. Such behavior of the nonlinear gains can be obtained by using the nonlinear function proposed in [Han, 1994].

The proposed enhanced DCAL control law with nonlinear feedback gains can be obtained by replacing the constant feedback gains in (3.3) by nonlinear time-varying ones as follows [Bennehar et al., 2014a, 2015b]:

$$\Gamma(t) = Y(q_d, \dot{q}_d, \ddot{q}_d)\hat{\Phi}(t) + K_P(.)e(t) + K_V(.)e_v(t) + \sigma \left\| e(t) \right\|^2 e_v(t) \tag{3.11}$$

where $K_P(.)$ and $K_V(.)$ are nonlinear feedback. The nonlinear gains in the case of PKMs are usually taken diagonal, which means that no coupling between the actuators is considered. Hence, $K_P(.)$ and $K_V(.)$ in (3.11) can be expressed as follows

$$K_P(.) = \text{diag}\left(k_{p_1}(.), k_{p_2}(.), \ldots, k_{p_n}(.)\right), \qquad K_V(.) = \text{diag}\left(k_{v_1}(.), k_{v_2}(.), \ldots, k_{v_n}(.)\right) \tag{3.12}$$

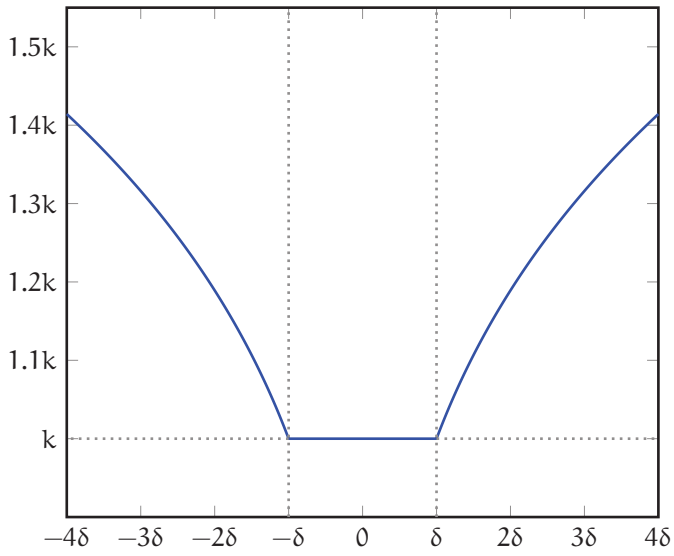


Figure 3.3 – Evolution of the proposed nonlinear gains versus the tracking error

Similar to Shang [Shang and Cong, 2009], we propose to use the nonlinear gains proposed in [Han, 1994] which satisfy our requirements. The $k_{p_i}(.)$ and $k_{v_i}(.)$ elements in (3.12) in this case are expressed as

$$k_{p_i}(e_i, \alpha_p, \delta_p) = \begin{cases} k_{p_0} |e_i(t)|^{\alpha_p - 1}, & |e_i(t)| > \delta_p \\ k_{p_0} \delta_p^{\alpha_p - 1}, & |e_i(t)| \leq \delta_p \end{cases} \tag{3.13a}$$

$$k_{v_i}(r_i, \alpha_v, \delta_v) = \begin{cases} k_{v_0} |r(t)|^{\alpha_v - 1}, & |r_i(t)| > \delta_v \\ k_{v_0} \delta_v^{\alpha_v - 1}, & |r_i(t)| \leq \delta_v \end{cases} \tag{3.13b}$$

where $k_{p_0}, k_{v_0}, \alpha_p, \delta_p, \alpha_v, \delta_v$ are positive design parameters to be tuned to get the desired performance. If the control design parameters α_p and α_v are chosen such that $\alpha_p > 1$ and $\alpha_v > 1$, then the following bounds hold globally for the nonlinear gains $K_P(.)$ and $K_V(.)$ given by (3.13a) and (3.13b), respectively:

$$0 < K_{P_m} \triangleq k_{p_0} \delta_p^{\alpha_p - 1} \mathbb{I}_n \leq K_p(.) \leq k_{p_0} \left(\delta_p^{\alpha_p - 1} + \|e\|_\infty^{\alpha_p - 1}\right) \mathbb{I}_n \triangleq K_{P_M} \tag{3.14}$$

$$0 < K_{V_m} \triangleq k_{v_0} \delta_v^{\alpha_v - 1} \mathbb{I}_n \leq K_v(.) \leq k_{v_0} \left(\delta_v^{\alpha_v - 1} + \|r\|_\infty^{\alpha_v - 1}\right) \mathbb{I}_n \triangleq K_{V_M} \tag{3.15}$$

with $\mathbb{I}_n$ being a $n \times n$ identity matrix and $\|.\|_\infty$ denotes the infinity-norm.

A typical example of the evolution of the proposed nonlinear gains versus time is shown in Figure 3.3.

Parameters adaptation law

To preserve the advantages of DCAL in terms of robustness and computation efficiency, the parameters adaptation law is kept unchanged and is given by (3.4).

Hence, the proposed extended controller shares the same advantages with the original controller, since the compensation of the nonlinearities and the adaptation of the parameters are both based on the desired quantities q_d, $\dot{q}_d$ and $\ddot{q}_d$ instead of the actual noisy or maybe unmeasurable ones.

3.2.4 Expected improvements w.r.t. original DCAL

Conventional controllers, being non-adaptive or adaptive, share the common drawback of using traditional fixed-gains linear PD feedback loops to eliminate the tracking error. However, the linear PD algorithm is known by the lack of robustness towards uncertainties and external disturbances. Consequently, several conventional controllers have been extended by replacing the fixed PD gains by nonlinear time-varying ones. Indeed, as it has been discussed in section 2.4.2, extended non-adaptive model-based controllers with nonlinear feedback gains applied to parallel manipulators provided significant improvements. A major outcome of these strategies is to improve the tracking performance of the closed-loop system.

Improved tracking

Motivated by the obtained results in [Shang and Shuang, 2009; Shang et al., 2009; Shang and Cong, 2009], where conventional non-adaptive controllers have been enhanced by nonlinear feedback gains, the proposed controller in (3.11) is expected to provide better tracking performances than the original DCAL in (3.3). For instance, in [Shang and Shuang, 2009], the augmented PD controller has been enhanced by using nonlinear feedback gains (NAPD) instead of the fixed ones. Experimental results demonstrate that the NAPD enhances the tracking errors by up to 39 %. Another example can be found in [Shang and Cong, 2009] where the CT controller is enhanced in a similar way to the APD. The proposed NCT controller based on nonlinear time-varying gains provides superior tracking performance than the conventional CT with constant feedback gains. Real-time experiments on a 2-DOF RA-PKM showed that the tracking performance with the proposed NCT controller were better than the original one.

Improved robustness

One of the main advantages of nonlinear feedback gains is their robustness towards uncertainties , external disturbances and variations of the dynamics and the environment. This is a desirable feature for PKMs since uncertainties are abundant in such systems. Especially in adaptive control, where the evolution of the estimated parameters holds two

phases; a transient and a steady-state phases. During the transient phase, the parameters evolve starting from their initial values toward their best steady-state ones. The estimated parameters may slowly converge or oscillate around their steady-state values according to the speed of adaptation. In this situation, the system is highly nonlinear and a robust feedback loop is required to cope with random behavior of the adaptive-dynamics-based loop.

Moreover, the adaptive loop in model-based adaptive controllers only handle the affine-in-the-parameters dynamics considered during the controller design. Any other factors, not considered in the adaptation mechanism, should be handled by the feedback loop. Consequently, a robust feedback algorithm is naturally required.

3.2.5 Tuning of the nonlinear gains

To tune the design parameters in the nonlinear gains (3.13a) and (3.13b), we follow the procedure proposed by Shang [Shang and Shuang, 2009] with some modifications to meet our needs. The proposed tuning method in [Shang and Shuang, 2009] enables the following behavior of the nonlinear gains:

- Position error gain $K_P(.)$: *large* gain for *small* error and *small* gain for *large* error.
- Velocity error gain $K_D(.)$: *large* gain for *large* error and *small* gain for *small* error.

However, early real-time experiments that have been performed on our platforms showed that this choice can be altered to improve the performance further. Thus, in our proposed controller, the following choice provides better results:

- Position error gain $K_P(.)$: *large* gain for *large* error and *small* gain for *small* error.
- Combined position-velocity error gain $K_V(.)$: *large* gain for *large* error and *small* gain for *small* error.

In order to tune the controller and to obtain the best design parameters for the nonlinear feedback gains, the following five-step procedure is followed

- **Step 1:** choose $\alpha_p = 1$ and $\alpha_v = 1$ and tune k_p and k_v using classical tuning methods of linear PD controllers.
- **Step 2:** keep the values of k_p and k_v obtained in the first step and measure the maximum error and combined error under this setting.
- **Step 3:** select the half of the maximum error as the value of δ_p and the half of the maximum combined error as the value of δ_v.
- **Step 4:** increase the value of α_p ($1 \leq \alpha_p \leq 1.5$) and decrease the value of k_p at the same time to improve the tracking performance.
- **Step 5:** increase the value of α_v ($1 \leq \alpha_v \leq 1.5$) and decrease the value of k_v at the same time to further improve the accuracy.

3.3 Contribution 2: RISE-Based Adaptive Control

RISE (an acronym for **R**obust **I**ntegral of the **S**ign of the **E**rror) is a recently developed robust nonlinear high gain feedback control strategy including a unique term based on the *signum* function [Xian et al., 2004]. RISE control law can accommodate for a large class of sufficiently smooth bounded disturbances. The main advantage of RISE is that stability of the closed-loop system is guaranteed, based on limited assumptions on the dynamics of the system, its environment and the external disturbances acting on it. Indeed, the employed *Lyapunov*-based proof of stability of RISE does not include restrictive assumptions about the system to be controlled. Moreover, RISE has the advantage of generating a continuous control law, compared to other robust controllers that can be found in the literature (e.g. variable structure [Zinober, 1994], sliding mode [Perruquetti and Barbot, 2002], etc.). This is very important when it comes to practical implementation of the controller since it avoids chattering effects caused by discontinuous control signals.

3.3.1 Background on RISE Feedback Control

In what follows, the problem formulation and the theoretical developments behind RISE control are explained. For the sake of simplicity, a first-order Single-Input-Single-Output (SISO) scalar system is considered. The extension to higher-order Multiple-Input-Multiple-Output (MIMO) systems is straightforward and does not involve complex modifications of the control law with respect to the first-order case. Afterwards, we explain how RISE can be applied on PKMs and enumerate some extensions and applications of RISE that can be found in the literature. At last, we present our contribution to improve the original RISE strategy.

Problem formulation

Consider the class of SISO nonlinear systems with dynamics of the following form

$$m(x)\dot{x} + f(x) = u(t) \tag{3.16}$$

where

$x(t) \in \mathbb{R}$ is the system state.

$u(t) \in \mathbb{R}$ is the control input.

$m(t), f(t) \in \mathbb{R}$ are uncertain nonlinear functions.

Let $x_d(t) \in \mathbb{R}$ be the reference trajectory to be tracked by the system state. The control objective is to achieve asymptotic tracking of the reference trajectory $x_d(t)$ by a continuous control law, despite the uncertainties in the dynamics (3.16).

Hypotheses and assumptions

The following hypotheses are essential to ensure the stability of RISE controlled system [Xian et al., 2004]

Assumption 1 *The uncertain nonlinear function* $m(x)$ *is positive and bounded from below by a positive constant* $\underline{m} \in \mathbb{R}$ *and from above by a positive non-decreasing function* $\overline{m}(x) \in \mathbb{R}$.

Assumption 2 *The reference trajectory* $x_d(t)$ *is designed such as it is continuously differentiable up to to its third derivative (i.e.* $x_d^{(i)}(t) \in \mathbb{R}$ $(i = 0,1,2,3)$ *exist and are bounded).*

Assumption 3 *The nonlinear functions* $m(x)$ *and* $f(x)$ *are twice differentiable with respect to the state variable* $x(t)$ *and their partial derivatives,* $\frac{\partial m(x)}{\partial x}, \frac{\partial^2 m(x)}{\partial x^2}, \frac{\partial f(x)}{\partial x}$ *and* $\frac{\partial^2 f(x)}{\partial x^2}$ *are bounded.*

Control law

Consider the tracking error $e(t)$ to be the difference between the reference and the actual system state at time $t \in \mathbb{R}^+$ as follows:

$$e(t) = x_d(t) - x(t) \tag{3.17}$$

The RISE feedback control law for the uncertain nonlinear system in (3.16) is given by [Xian et al., 2004]:

$$u(t) = (k_s + 1)e(t) - (k_s + 1)e(t_0) + \int_{t_0}^{t} [(k_s + 1)\alpha e(\sigma) + \beta \operatorname{sgn}(e(\sigma))]\, d\sigma \tag{3.18}$$

where

$k_s, \alpha, \beta \in \mathbb{R}^+$ are positive control design parameters.

t_0 is the initial time.

$\operatorname{sgn}(.)$ denotes the standard signum function.

The second term in the control law (i.e. $(k_s + 1)e(t_0)$) is only needed to ensure that the control input at $t = t_0$ is zero (i.e. $u(t_0) = 0$).

The control law (3.18) ensures that the system asymptotically tracks the reference trajectory, provided that the control gains k_s and β are chosen large enough with respect to the norm of the initial error $e(t_0)$ and some bound based on the reference trajectory. For the full stability analysis, the reader is referred to [Xian et al., 2004].

As it can be seen, RISE control law (3.18) is only based on the state of the system. The only required information is the tracking error.

3.3.2 RISE for higher-order MIMO systems

The extension of RISE to higher order Multiple-Input-Multiple-Output nonlinear systems is quite straightforward. In fact, this problem has been addressed in the original paper where RISE has been described for the first time [Xian et al., 2004].

Consider the n^{th} order MIMO uncertain nonlinear system given by the following expression

$$M(x,\dot{x},\ldots,x^{(n-1)})x^n + F(x,\dot{x},\ldots,x^{(n-1)}) = U(t) \tag{3.19}$$

where

$X \triangleq [x,\dot{x},\ldots,x^{(n-1)}]^T \in \mathbb{R}^m$ is the state vector.

$U(t) \in \mathbb{R}^m$ is the control input vector.

$M(x,\dot{x},\ldots,x^{(n-1)}) \in \mathbb{R}^{m\times m}$ and $F(x,\dot{x},\ldots,x^{(n-1)}) \in \mathbb{R}^m$ are uncertain nonlinear functions.

Hypotheses and assumptions

To fulfill the requirements of the stability proof [Xian et al., 2004], the following assumptions are considered

Assumption 4 *The uncertain nonlinear system matrix $M(.)$ is bounded from below by a positive constant $\underline{m} \in \mathbb{R}$, and from above by a nondecreasing positive function $\overline{m}(x) \in \mathbb{R}$. Consequently, the following inequality holds:*

$$\underline{m}\|\xi\|^2 \leq \xi^T M(.)\xi \leq \overline{m}(x)\|\xi\|^2, \quad \forall \xi \in \mathbb{R}^n. \tag{3.20}$$

Assumption 5 *The nonlinear functions $M(.)$ and $F(.)$ are second-order differentiable.*

Assumption 6 *The desired reference trajectory to be tracked by the system is continuously differentiable with respect to time, up to the $(n+2)$th derivative.*

Control law

Since the system is high-order, we start by defining the following auxiliary tracking error system:

$$\begin{aligned}
e_1(t) &\triangleq x_d(t) - x(t) \\
e_2(t) &\triangleq \dot{e}_1(t) + e_1(t) \\
e_3(t) &\triangleq \dot{e}_2(t) + e_2(t) + e_1(t) \\
e_4(t) &\triangleq \dot{e}_3(t) + e_3(t) + e_2(t) \\
&\vdots \\
e_i(t) &\triangleq \dot{e}_{i-1}(t) + e_{i-1}(t) + e_{i-2}(t) \\
&\vdots \\
e_n(t) &\triangleq \dot{e}_{n-1}(t) + e_{n-1}(t) + e_{n-2}(t)
\end{aligned} \tag{3.21}$$

Based on the stability analysis, the control law is designed as follows [Xian et al., 2004]:

$$U(t) = (K_s + I_m)e_n(t) - (K_s + I_m)e_n(t_0) + \int_{t_0}^{t} [(K_s + I_m)Ae_n(\sigma) + B\,\mathrm{sgn}(e_n(\sigma))]\,d\sigma \tag{3.22}$$

where I_m is an identity matrix of dimension $m \times m$. As it can be seen, the control law is very similar to the SISO case with few modifications. First, the tracking error $e(t)$ has been replaced by the combined one $e_n(t)$. Second, since the combined tracking error is now an m-dimensional vector, the controller design parameters are no longer scalars. Thus, K_s, A and B are in this case $(m \times m)$ positive-definite, diagonal gain matrices.

3.3.3 RISE control of parallel manipulators

Let's recall the dynamics of a m-DOF parallel manipulator, subject to external disturbances and friction effects, given in joint space as follows

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + \Gamma_d(t) = \Gamma(t) \tag{3.23}$$

which can be put in a similar form to (3.19) by gathering all non-inertial effects as follows:

$$M(q)\ddot{q} + F(q,\dot{q},t) = \Gamma(t) \tag{3.24}$$

where $F(q,\dot{q},t) \triangleq C(q,\dot{q})\dot{q} + G(q) + f(q,\dot{q}) + \Gamma_d(t)$ is a n-dimensional vector gathering all the non-inertial effects in the PKM. The nonlinear dynamics (3.24) is clearly a 2^{nd} order nonlinear system with n-dimensional input vector $\Gamma(t)$.

In a practical situation, where the robot should be in an industrial environment executing various tasks, the matrices $M(.)$ and $F(.)$ are likely to vary. Furthermore, the robot may

be subject to various disturbances and uncertain factors such as friction and interaction with the environment. Hence, it is of great importance to design a control law that takes into consideration all these variations and uncertainties.

To quantify the control objective, the joint tracking error $e_1(t) \in \mathbb{R}^n$ and the filtered tracking error $e_2(t) \in \mathbb{R}^n$ can be defined as follows

$$e_1(t) = q_d(t) - q(t) \tag{3.25}$$
$$e_2(t) = \dot{e}_1(t) + \alpha_1 e_1(t), \quad \alpha_1 > 0 \tag{3.26}$$

being $q_d \in \mathbb{R}^n$ the desired joint position. Notice the introduction of the weighting parameter α_1 allows us to have an additional flexibility when tuning the controller. The RISE control for systems whose dynamics are governed by (3.24) is expressed as follows:

$$\Gamma_{RISE} = (k_s+1)e_2 - (k_s+1)e_2(t_0) + \int_{t_0}^{t} [(k_s+1)\alpha_2 e_2(\sigma) + \beta\,\mathrm{sgn}(e_2(\sigma))]\,d\sigma \tag{3.27}$$

where $k_s, \alpha_2, \beta \in \mathbb{R}$ are positive constant gains and $\mathrm{sgn}(\xi)$ is defined $\forall \xi = [\ \xi_1 \quad \xi_2 \quad \dots \quad \xi_n\ ] \in \mathbb{R}^n$ as $\mathrm{sgn}(\xi) \triangleq [\ \mathrm{sgn}(\xi_1) \quad \mathrm{sgn}(\xi_2) \quad \dots \quad \mathrm{sgn}(\xi_n)\ ]$.

3.3.4 Some extensions and applications of RISE

RISE control has been adopted and successfully applied in various fields including robotics. In the following, the most relevant extensions and applications of RISE proposed in the literature are enumerated. In [Fischer et al., 2014], RISE has been experimentally validated on a fully-actuated autonomous underwater vehicle (AUV) in both controlled and open-water environments. The use of RISE in this work was motivated by the desire to obtain a suitable tracking performance despite the abundance of uncertainties and disturbances in such systems (e.g. model uncertainties, current, waves, etc.). Even though the obtained results in open-water were not as good as those in controlled environment, the tracking performance was within the desired range.

RISE has been augmented with a model-based adaptive term in [Patre et al., 2008] to control a direct-drive motor. This choice is motivated by the desire to include some knowledge of the dynamics in the control loop to improve the tracking performance and to reduce the control effort. The dynamics of the motor has been separated into structured (linearly parameterizable) and unstructured uncertainties. Consequently, the structured uncertainties will be accounted for by the added adaptive term, while the remaining uncertainties are compensated for by means of the RISE term. However, the extension of this work to PKMs may not be investigated due to the proposed adaptation law. In fact, the evolution of the estimated parameters depends on the first and second time derivatives of the regression matrix $Y(.)$. This can be a tedious task for parallel manipulators since it is very time consuming and cannot be computed in real-time.

In [Taktak-Meziou et al., 2015], RISE has been combined with a neural network-based (NN) feedforward term (RISE-NN) to control a read/write head of a hard disk drive (HDD). The use of neural networks as a feedforward control term is motivated by their universal approximation property. The additional feedforward term estimates the disturbances that act on the system in real-time in order to cancel them and achieve the control objective. The provided simulation results showed that the proposed augmented RISE controller provided better tracking performance and less control effort than a classical PD controller. This work has been further extended by considering a prediction-based optimization to the RISE-NN strategy (P-RISE-NN). The proposed strategy allowed to determine in real-time the best value of the feedback parameter k_s. The provided simulation results demonstrated the relevance of the proposed solution compared to the original RISE-NN.

3.3.5 Proposed contribution: RISE-based adaptive control for PKMs

To improve the overall closed-loop performance of RISE control for parallel manipulators, one solution would be to include the modeled dynamics in the control loop. As it was reported in [Patre et al., 2008], the combination of the modeled dynamics with RISE control yields better tracking performance and reduced control effort. Unfortunately, the proposed adaptation law in this work was too complicated to be implemented on PKMs. Hence, an alternative adaptation law should be considered.

The most practical strategy is to consider including a model-based feedforward term in the control loop for partial compensation of the nonlinear dynamics. Such strategy has demonstrated its effectiveness compared to exact compensation architectures. To that aim, consider the dynamics of a m-DOF PKM in joint space given by (3.24). This dynamics can be rearranged into structured and unstructured uncertainties. Structured uncertainties adhere to the affine-in-the-parameters property, while unstructured uncertainties do not necessarily obey to that property. Hence, the dynamics of the parallel manipulator can be rewritten as:

$$Y(q,\dot{q},\ddot{q})\Phi + \Gamma_d(t) = \Gamma(t). \tag{3.28}$$

Structured uncertainties are mainly due to uncertainties and variations in the parameters of the robot while operating. However, external disturbances are generally unstructured and are known by their hazardous and unpredictable behavior.

Now, consider the following control law consisting of an augmented RISE with nominal linear-in-the-parameters feedforward dynamics [Bennehar et al., 2014b]:

$$\begin{aligned}\Gamma(t) =& Y(q_d,\dot{q}_d,\ddot{q}_d)\Phi + (k_s+1)e_2 - (k_s+1)e_2(t_0) \\ &+ \int_{t_0}^{t} \left[(k_s+1)\alpha_2 e_2(\sigma) + \beta\,\mathrm{sgn}(e_2(\sigma))\right] d\sigma\end{aligned} \tag{3.29}$$

where the included feedforward $Y(q,\dot{q},\ddot{q})\Phi$ is only based on the modelled dynamics, the desired trajectories and the best know value of Φ. Consequently, part of the nonlinear dy-

namics will be handled by the additional model-based term. Also, errors emerging from the simplifying hypotheses and non-modeled phenomena are not included in the feedforward term. Therefore, in addition to the external disturbances, they should be handled by the RISE term.

If the additional model-based feedforward matches the actual dynamics of the manipulator, the control law in (3.29) is expected to yield improved tracking compared to original RISE controller. However, if the dynamic parameters are uncertain, unknown or time-varying, the compensation may not be precise. Therefore, one should consider to readjust the model-based loop in real-time. Hence, the nominal augmentation term in (3.29) is replaced by an adaptive one as follows:

$$\begin{aligned}\Gamma(t) =& Y(q_d, \dot{q}_d, \ddot{q}_d)\hat{\Phi}(t) + (k_s+1)e_2 - (k_s+1)e_2(t_0) \\ &+ \int_{t_0}^{t} [(k_s+1)\alpha_2 e_2(\sigma) + \beta\,\mathrm{sgn}(e_2(\sigma))]\,d\sigma\end{aligned} \tag{3.30}$$

where the nominal parameters vector Φ is replaced by its estimate $\hat{\Phi}(t)$ that should be updated in real-time. Since we envision to implement the proposed controller in real-time, we should consider a simple adaptation law that do not require heavy computations. Hence, we propose the following adaptation law:

$$\dot{\hat{\theta}}_i(t) = \begin{cases} k_i\psi_i, & \text{if } -\phi_{b_i} < \hat{\phi} < \phi_{b_i} \text{ or} \\ & \hat{\phi} \geq \phi_{b_i} \text{ and } \phi_i \leq 0 \text{ or} \\ & \hat{\phi} \leq -\phi_{b_i} \text{ and } \phi_i \geq 0 \\ 0, & \text{if } \hat{\phi} \geq \phi_{b_i} \text{ and } \phi_i \geq 0 \text{ or} \\ & \hat{\phi} \leq -\phi_{b_i} \text{ and } \phi_i \leq 0 \end{cases} \tag{3.31}$$

where

- $\hat{\phi}_i$ is the estimate of the unknown parameter ϕ_i,
- $k_i > 0$ is the i^th element of the diagonal adaptation matrix $K \triangleq \mathrm{diag}(k_1, k_2, \ldots, k_p) \in \mathbb{R}^{p\times p}$,
- ψ_i is the i^th element of the column matrix $\psi \triangleq Y_d^T(\cdot)e_2$,
- ϕ_{b_i} is a bound of the estimated parameter $\hat{\phi}_i$.

The adaptation law (3.31) ensures that the estimated parameters $\hat{\Phi}(t) \triangleq [\hat{\phi}_1(t), \hat{\phi}_2(t), \ldots, \hat{\phi}_p(t)]^T \in \mathbb{R}^p$ remain within allowable range, i.e. $-\Phi_b \leq \left\|\hat{\Phi}(t)\right\|_\infty \leq \Phi_b, \forall t \geq 0$. The adaptation law (3.31) is as simple as it can get since it does not involve any time derivative of the regression matrix . Moreover, its elements are evaluated using the desired values of the joint positions, velocities and accelerations.

3.3.6 Expected improvements w.r.t. original RISE

The proposed control law in (3.30) consists of two distinct control elements. The first element is the model-based adaptive term which is responsible of the compensation of the linear-in-the-parameters uncertainties. This compensation, however, is only partial and not exact since it is based on desired trajectories and not measured ones. The second element is the RISE feedback term that accommodates for the remaining dynamics and possible disturbances acting on the system.

The motivation behind the proposed strategy is mainly to achieve better tracking performance. The main reason for this is the addition of the knowledge about the structure of the dynamics in the control loop. Indeed, according to what has been concluded in [Patre et al., 2008], the more the uncertainty is unstructured the more the performance of the system is diminished. Then if the structure of part or all the dynamics is known, it should be more adequate to include it in the control loop to improve the performance.

Another outcome that can be achieved through the proposed controller is the diminution of the control efforts and the avoidance of high-gain feedback [Patre et al., 2008]. Such feature reduces the energy consumption and enables the manipulator to reach higher accelerations without violating its actuators' limits.

Finally, compared to the prior proposed DCAL-based strategy, the adaptive RISE controller may improve the robustness of the closed-loop system. In fact, RISE feedback term can be considered as a nonlinear feedback loop with verified robustness outcomes.

3.4 Conclusion

In this chapter, two model-based adaptive control schemes for PKMs were proposed. The first proposed control solution is based on the well-known DCAL, commonly used in robotics. After analyzing the original architecture of DCAL, it was noticed that, besides the adaptive compensation loop, a linear PD feedback control term with constant gains is used to ensure the tracking of desired trajectories. To improve the tracking capabilities, we have proposed to replace the standard PD feedback loop with a nonlinear time-varying one. Hence, the proposed controller holds the simplicity of DCAL and the efficiency of nonlinear feedback gains.

The second proposed solution consists of a novel adaptive control scheme based on a new robust nonlinear feedback strategy called RISE. The main outcome of RISE is its ability to account for a large class of general nonlinear uncertainties acting on the system based on very limited assumptions. The proposed control solution involves the combination of RISE feedback and model-based adaptive control. The proposed solution is motivated by the fact that the uncertainties of parallel manipulators can be divided into structured and unstructured ones. The augmentation of RISE by a model-based adaptive control term

enables better compensation of structured uncertainties to enable better tracking capabilities.

Both of the above proposed strategies issued from model-based adaptive control. In the next chapter, we focus on a new adaptive control strategy based on the well-known Model Reference Adaptive Control, which is conceptually different from what we have seen so far.

CHAPTER

4 Solution 2: Extended $\mathcal{L}_1$ adaptive control

Contents

4.1 Introduction

This chapter is dedicated to the description of the proposed control strategies for parallel manipulators based on $\mathcal{L}_1$ adaptive control.

Chronologically speaking, the first adaptive control strategies developed for mechanical mechanical manipulators were based on Model Reference Adaptive Control (MRAC). These controllers hold the advantage of not relying on any knowledge of the dynamics of the manipulator. However, several shortcomings can be noticed in MRAC-based schemes that limited their spreading in the robotics community [Anderson, 2005]. One of the main drawbacks that may cause the failure of a MRAC scheme is the coupling between the control and adaptation loops. This is mainly due to the fact that the adaptive gain in MRAC acts as a feedback gain, meaning that a trade-off has to be made between the speed of adaptation and the robustness of the controller. Recently, a new strategy called $\mathcal{L}_1$ adaptive control has been developed in order to overcome this problem [Chengyu and Hovakimyan,

2006a,b]. The structure of $\mathcal{L}_1$ adaptive control features a unique filtering technique that enables a decoupling between the estimation and the control loops. This means that the adaptation can be made arbitrarily fast while guaranteeing sufficient robustness margins.

With the advances in modeling techniques, obtaining an accurate dynamic model for a robot manipulator became relatively an easy task. Consequently, model-based controllers that take advantage of the dynamics knowledge in the control design in the aim of improving the control performance captured an increasing attention. In this context, we will explain how $\mathcal{L}_1$ adaptive control and model-based control can be combined to improve the performance of the control system.

4.2 Background on $\mathcal{L}_1$ adaptive control

In this section, the basic idea behind the development of the $\mathcal{L}_1$ adaptive control theory is detailed. For the sake of simplicity, we consider a single-input single-output system. Since $\mathcal{L}_1$ adaptive control is inspired from MRAC, we first introduce MRAC for the considered system. Then the architecture of MRAC is changed to achieve the decoupling of robustness and adaptation.

4.2.1 Control problem formulation

Consider the following single-input single-output linear time-invariant system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + bu(t), \quad x(0) = x_0 \\ y(t) &= c^T x(t) \end{aligned} \tag{4.1}$$

where

$x(t) \in \mathbb{R}^n$ is the measurable state of the system,

$u(t) \in \mathbb{R}$ is the control input,

$b, c \in \mathbb{R}^n$ are known constant input and output vectors,

$y(t) \in \mathbb{R}$ is the output to be regulated by the controller.

The system state matrix $A \in \mathbb{R}^{n\times n}$ is supposed unknown. The control objective is to design an adaptive control signal $u(t)$ such that the regulated output of the system $y(t)$ tracks a given reference signal $r(t) \in \mathbb{R}$ with desired performance while guaranteeing that the system state and other signals remain bounded under the following assumption

Assumption 7 *There exist a Hurwitz matrix $A_m \in \mathbb{R}^{n\times n}$ and a vector $\theta \in \mathbb{R}^n$ of perfect parameters such that the pair (A_m, b) is controllable and $A_m - A = b\theta^T$. Moreover, assume that this unknown perfect parameters vector θ belongs to a given compact set Θ.*

Under the above assumption, the system in (4.1) can be rewritten as follows

$$\begin{aligned} \dot{x}(t) &= A_m x(t) + b\left(u(t) - \theta^T x(t)\right), \quad x(0) = x_0 \\ y(t) &= c^T x(t) \end{aligned} \tag{4.2}$$

4.2.2 From direct MRAC to direct MRAC with a state predictor

One of the key differences between MRAC and $\mathcal{L}_1$ adaptive control is the introduction of a prediction-based adaptive architecture. In the following, this new structure is highlighted and demonstrated that it leads to the same closed-loop behavior as direct MRAC.

Direct MRAC

Consider the nominal nonadaptive controller given by

$$u_{nom}(t) = \theta^T x(t) + k_g r(t) \tag{4.3}$$

where $k_g \triangleq 1/\left(c^T A_m^{-1} b\right)$ is a feedforward gain that ensures zero steady-state tracking error. Substituting (4.3) in the system dynamics in (4.2) leads to the following desired reference system

$$\begin{aligned} \dot{x}_m(t) &= A_m x_m(t) + b k_g r(t), \quad x(0) = x_0 \\ y_m(t) &= c^T x_m(t) \end{aligned} \tag{4.4}$$

where $x_m(t)$ is the state of the reference system and $y_m(t)$ its output. The nominal control law is impossible to implement due to the uncertainties in the system state matrix A (and, hence, θ) which should be estimated online.

The feasible direct MRAC control law is hence given by

$$u(t) = \hat{\theta}^T x(t) + k_g r(t) \tag{4.5}$$

where $\hat{\theta} \in \mathbb{R}^n$ is an estimate of θ whose evolution is governed by the following adaptation rule

$$\dot{\hat{\theta}}(t) = \gamma x(t) e^T(t) P b, \quad \hat{\theta}(0) = k_{x0} \tag{4.6}$$

where $\gamma \in \mathbb{R}^+$ is the adaptation gain, $e(t) \triangleq x_m(t) - x(t)$ and $P = P^T > 0$ is the solution for the algebraic Lyapunov equation $A_m^T P + P A_m = -Q$, for an arbitrary choice of $Q = Q^T > 0$.

Substituting the adaptive control law (4.5) in the system dynamics in (4.2) leads to the following error dynamics

$$\dot{e}(t) = A_m e(t) - b\tilde{\theta}^T(t) x(t) \tag{4.7}$$

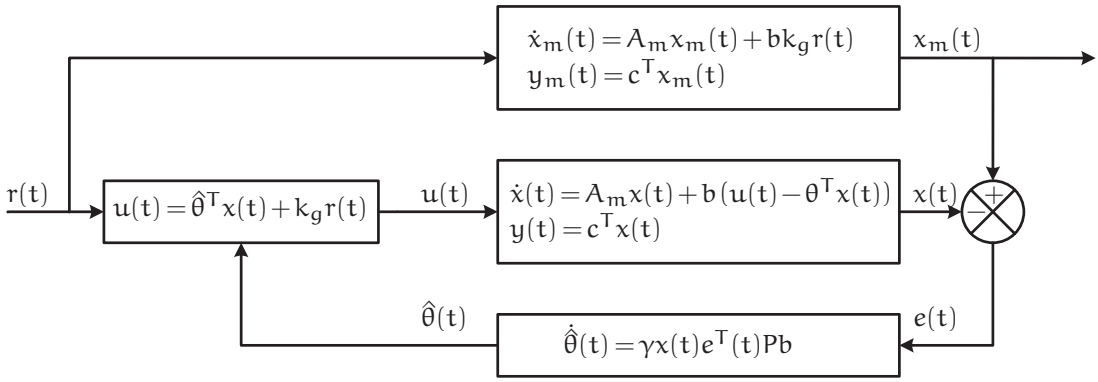


Figure 4.1 – Block diagram of MRAC

where $\tilde{\theta} \triangleq \hat{\theta}(t) - \theta$ is the parameters estimation error. Figure 4.1 depicts the overall block diagram of MRAC.

MRAC with a state predictor

Now consider the following state predictor that mimics the behavior of the system in (4.2) with the unknown parameter θ replaced by its estimate $\hat{\theta}(t)$

$$\begin{aligned} \dot{\hat{x}}(t) &= A_m \hat{x}(t) + b(u(t) - \hat{\theta} x(t)), \quad \hat{x}(0) = x_0 \\ \hat{y}(t) &= c^T \hat{x}(t) \end{aligned} \tag{4.8}$$

where $\hat{x}(t)$ being the state of the predictor. The prediction error dynamics can be obtained by subtracting the system dynamics from those of the predictor as follows

$$\dot{\tilde{x}}(t) = A_m \tilde{x}(t) - b\tilde{\theta}(t) x(t) \tag{4.9}$$

where $\tilde{x}(t) \triangleq \hat{x}(t) - x(t)$ is the prediction error. It can be seen that the prediction error dynamics is identical to the obtained error dynamics with direct MRAC in (B.4). The adaptive law of the predictor-based controller is similar to that of direct MRAC, the only difference is the use of the prediction error $\tilde{x}(t)$ instead of the tracking error $e(t)$. It is thus given by

$$\dot{\hat{\theta}}(t) = \gamma x(t) \tilde{x}^T(t) P b, \quad \hat{\theta}(0) = \theta_0 \tag{4.10}$$

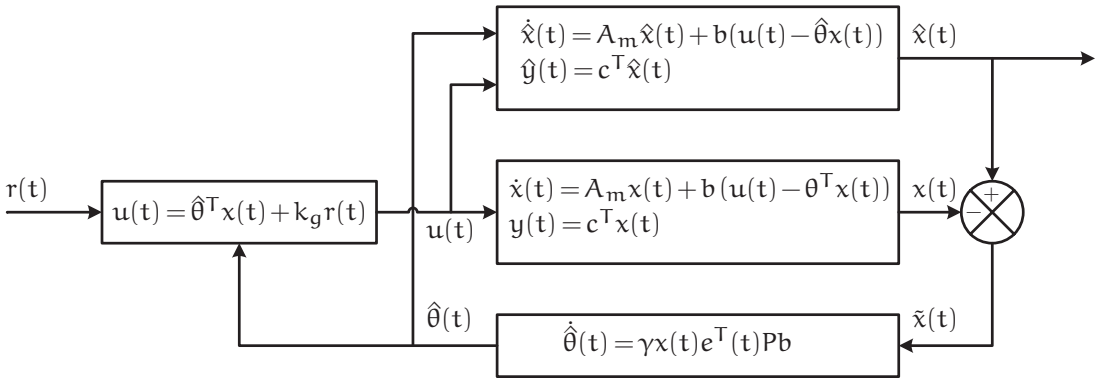


Figure 4.2 – Block diagram of the control loop based on MRAC with state predictor

It can be noticed that the closed-loop state predictor mimics the behavior of the reference model in (4.4). Indeed, substituting the control law (4.5) in the predictor dynamics and upon the use of (4.10) we get the following closed-loop dynamics

$$\begin{aligned} \dot{\hat{x}}(t) &= A_m\hat{x}(t) + bk_g r(t), \quad \hat{x}(0) = x0, \\ \hat{y}(t) &= c^T\hat{x}(t) \end{aligned} \tag{4.11}$$

which is identical to the dynamics of the reference system of direct MRAC in (4.4). It is demonstrated in [Hovakimyan and Cao, 2010] that the tracking (or prediction) errors are upper bounded at any time by

$$\|e(t)\| (= \|\tilde{x}(t)\|) \leq \frac{\|\tilde{\theta}(0)\|}{\sqrt{\lambda_{min}(P)\gamma}} \tag{4.12}$$

being $\tilde{\theta}(0)$ the initial parameters estimation error and $\lambda_m in(P)$ the minimum eigenvalue of P. This means that the tracking error can be made arbitrarily small by increasing the adaptation gain γ. However, it can be seen from the control law in (4.5) and the adaptation laws in (4.6) and (4.10) that increasing the adaptation gain lead to high gain feedback. The block diagram of MRAC scheme with a state predictor is shown in Figure 4.2.

4.2.3 $\mathcal{L}_1$ adaptive control

In a real system, the unknown parameters vector θ may be a time-varying uncertainty with frequencies that may lie outside the control channel bandwidth [Hovakimyan and Cao, 2010]. In this case, such frequencies are naturally attenuated due to hardware limitations of the actuators. Consequently, the behavior of the closed-loop system will be different than the dynamics of the reference model in (4.4).

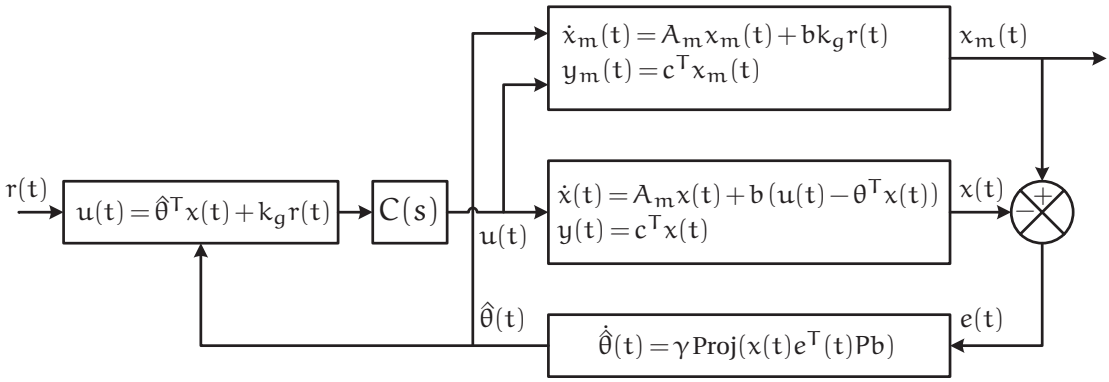


Figure 4.3 – Block diagram of $\mathcal{L}_1$ adaptive control.

The control problem in $\mathcal{L}_1$ adaptive control is formulated with the understanding that some uncertainties could never be perfectly compensated. Indeed, while the control objective in MRAC is only stated asymptotically, the closed-loop performance in $\mathcal{L}_1$ adaptive control is specified $\forall t \geq 0$. This means that at any time, the performance of the system can be predicted. Moreover, the control signal never exceeds the available control channel bandwidth.

For the class of systems given by (4.2), consider the following state predictor that mimics its behavior

$$\begin{aligned} \dot{\hat{x}}(t) &= A_m \hat{x}(t) + b(u(t) - \hat{\theta} x(t)), \quad \hat{x}(0) = x_0 \\ \hat{y}(t) &= c^T \hat{x}(t) \end{aligned} \tag{4.13}$$

In addition to the state predictor in (4.13), consider a projection-type adaption law for the estimated parameter vector $\hat{\theta}$ expressed as

$$\dot{\hat{\theta}}(t) = \gamma\,\mathrm{Proj}\,(\hat{\theta}(t), x(t)\tilde{x}(t)Pb), \quad \hat{\theta}(0) = \theta_0 \tag{4.14}$$

which is adjusted using the prediction error $\tilde{x}(t) = \hat{x}(t) - x(t)$. The projection operator Proj avoids the parameters drift and ensures that they remain inside the compact set Θ. In (4.14), γ is the positive adaptation gain and P is the symmetric positive definite matrix, solution of the algebraic *Lyapunov* equation $A_m^T P + P A_m = -Q$ for an arbitrary symmetric positive definite matrix Q.

The last stage which is one of the notable unique features of the $\mathcal{L}_1$ adaptive control, is the control input characterized by the introduction of a low-pass filter. It is given by its *Laplace* form as follows

$$u(s) = C(s)\,(\hat{\eta}(s) + k_g r(s)) \tag{4.15}$$

where $C(s)$ is a bounded-input bounded-output strictly proper transfer function with $C(0) = 1$, $r(s)$ is the *Laplace* transform of the reference trajectory $r(t)$ and $\hat{\eta}(s)$ is the *Laplace* transform of $\eta(\widehat{t}) = \widehat{\theta}^{\top}(t)x(t)$.

It is demonstrated in [Hovakimyan and Cao, 2010] that the controlled system in (4.1) under the $\mathcal{L}_1$ adaptive control law in (4.15) is bounded-input bounded-state with respect to $r(t)$ and x_0 if the following $\mathcal{L}_1$-norm-based equality is satisfied

$$\|G(s)\|_{\mathcal{L}_1} L < 1 \tag{4.16}$$

where

$$G(s) \triangleq H(s)(1 - C(s)), \quad H(s) \triangleq (s\mathbb{I} - A_m)^{-1} b, \quad L \triangleq \max_{\theta \in \Theta} \|\theta\|_1 \tag{4.17}$$

The block diagram of the $\mathcal{L}_1$ adaptive control strategy for the class of systems given by (4.2) is displayed in Figure 4.3.

4.3 $\mathcal{L}_1$ Adaptive control for mechanical manipulators

Recall the inverse dynamics for parallel mechanical manipulators which are expressed as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \Gamma_d(t) = \Gamma(t) \tag{4.18}$$

In the aim of developing a $\mathcal{L}_1$ adaptive controller for the class of systems given by (4.18), introduce the following combined position-velocity tracking error given by

$$r = (\dot{q} - \dot{q}_d) + \Lambda(q - q_d) \tag{4.19}$$

where $\Lambda \in \mathbb{R}^{4\times 4}$ is a symmetric positive-definite weighting matrix.

4.3.1 Control law

Consider the following control input vector $\Gamma(t)$ consisting of a combination of two distinct terms [Bennehar et al., 2015d]

$$\Gamma(t) = \Gamma_m(t) + \Gamma_{ad}(t), \quad \Gamma_m(t) \triangleq A_m r(t) \tag{4.20}$$

where $\Gamma_m \in \mathbb{R}^n$ is a state-feedback term characterizing the desired transient response of the tracking error $r(t)$ and Γ_{ad} is the adaptive control term which will be designed subsequently. Taking the first time derivative of (4.19) with respect to time we get

$$\dot{r}(t) = (\ddot{q} - \ddot{q}_d) + \Lambda(\dot{q} - \dot{q}_d) \tag{4.21}$$

Substituting the control law (4.20) into the dynamics of the parallel manipulator in (4.18) and solving for $\ddot{q}$ along with substituting into (4.21), we get the following error dynamics

$$\dot{r}(t) = A_m r(t) + \Gamma_{ad}(t) - \eta(t, \zeta(t)), \quad r(0) = r_0 \tag{4.22}$$

where $\zeta = [r, q]^T$ and $\eta(t, \zeta(t))$ is a nonlinear function that gathers all the nonlinearities of the system including possible external disturbances. Before proceeding to the development of the adaptive control law, some important assumptions and properties regarding the unknown function $\eta(t, \zeta(t))$ have to be made, they can be summarized as follows

Assumption 8 ***(Uniform boundedness of*** $\eta(t,0)$***)*** *There exist* $B > 0$ *such that* $\forall t \geq 0$, $\|\eta(t,0)\| \leq B$.

Assumption 9 ***(Semiglobal uniform boundedness of partial derivatives of*** $\eta(t,\zeta(t))$***)*** *The unknown nonlinear function* $\eta(t,\zeta(t))$ *is continuous with respect to its arguments, and for arbitrary* $\delta > 0$*, there exist* $d_{\eta_t}(\delta)$ *and* $d_{\eta_x}(\delta)$ *such that*

$$\left\| \frac{\partial \eta(t,\zeta)}{\partial t} \right\|_\infty \leq d_{\eta_t}(\delta), \quad \left\| \frac{\partial \eta(t,\zeta)}{\partial \zeta} \right\|_\infty \leq d_{\eta_\zeta}(\delta) \tag{4.23}$$

Assumption 10 *for* $t \geq 0$, $\|r_t\|_{\mathcal{L}_\infty} \leq \rho$ *and* $\|\dot{r}_T\|_{\mathcal{L}_\infty} \leq d_r$*, for some positive constants* ρ *and* d_r.

It follows from Lemma A.9.2 in [Hovakimyan and Cao, 2010] that the unknown nonlinear function $\eta(t,\zeta(t))$ can be parametrized as follows

$$\eta(t,\zeta(t)) = \theta(t)\|r_t\|_{\mathcal{L}_\infty} + \sigma(t) \tag{4.24}$$

where $\theta(t), \sigma(t) \in \mathbb{R}^n$ are continuous, piecewise-differentiable and uniformly bounded unknown functions. Therefore, the error dynamics in (4.22) can be rewritten as:

$$\dot{r}(t) = A_m r(t) + \Gamma_{ad}(t) - (\theta(t)\|r_t\|_{\mathcal{L}_\infty} + \sigma(t)), \quad r(0) = r_0. \tag{4.25}$$

Notice that the substitution of $\Gamma_{ad}(t) = (\theta(t)\|r_t\|_{\mathcal{L}_\infty} + \sigma(t))$ in the error dynamics in (4.25) results in the desired performance characterized by $\dot{r}(t) = A_m r(t)$. However, since the nonlinear function $\eta(t,\zeta(t)) = (\theta(t)\|r_t\|_{\mathcal{L}_\infty} + \sigma(t))$ is unknown due to uncertainties and unmeasured disturbances, the control term $\Gamma_{ad}(t)$ should be designed in a way that it estimates the unknown functions $\theta(t)$ and $\sigma(t)$ in real-time.

4.3.2 State predictor

To predict the behavior of the combined tracking error $r(t)$ and according to the $\mathcal{L}_1$ adaptive control theory, a state predictor that mimics the behavior of the error dynamics in (4.25) is formulated by [Bennehar et al., 2015d]:

$$\dot{\hat{r}}(t) = A_m \hat{r}(t) + \Gamma_{ad}(t) - (\hat{\theta}(t)\|r_t\|_{\mathcal{L}_\infty} + \hat{\sigma}(t)) - Z\tilde{r}(t), \quad \hat{r}(0) = r_0 \tag{4.26}$$

where $\hat{r}(t)$ is the state of the predictor, $\tilde{r}(t) \triangleq \hat{r}(t) - r(t)$ is the prediction error and $\hat{\theta}(t), \hat{\sigma}(t)$ are estimates of $\theta(t)$ and $\sigma(t)$, respectively. Notice the introduction of the additional term $Z\tilde{r}(t)$ responsible of rejecting the estimation error.

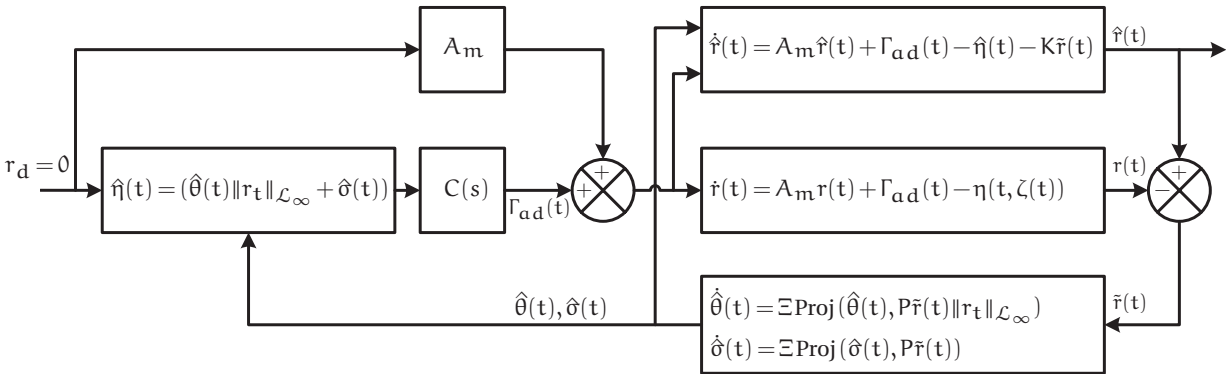


Figure 4.4 – Block diagram of $\mathcal{L}_1$ adaptive control for parallel manipulators

4.3.3 Adaptation laws

The evolution of the adaptive estimates $\hat{\theta}(t), \hat{\sigma}(t)$, required for the $\mathcal{L}_1$ adaptive control architecture, is governed by the following projection-type adaptation laws

$$\dot{\hat{\theta}}(t) = \Xi\text{Proj}(\hat{\theta}(t), P\tilde{r}(t)\|r_t\|_{\mathcal{L}_\infty}), \qquad \hat{\theta}(0) = \hat{\theta}_0 \tag{4.27}$$

$$\dot{\hat{\sigma}}(t) = \Xi\text{Proj}(\hat{\sigma}(t), P\tilde{r}(t)), \qquad \hat{\sigma}(0) = \hat{\sigma}_0 \tag{4.28}$$

where $\Xi > 0$ is the adaptive gain, $P = P^T > 0$ is the solution to the algebraic *Lyapunov* equation $A_m^T P + PA_m = -Q$ for some arbitrary matrix $Q = Q^T > 0$.

Finally, the adaptive control term $\Gamma_{ad}(t)$ in (4.20) is given by its *Laplace* form as follows

$$\Gamma_{ad}(s) = C(s)\hat{\eta}(s) \tag{4.29}$$

where $\hat{\eta}(s)$ is the Laplace transform of $\hat{\eta}(t) = (\hat{\theta}(t)\|r_t\|_{\mathcal{L}_\infty} + \hat{\sigma}(t))$ and $C(s)$ is a diagonal matrix whose elements are bounded-input bounded-output stable strictly proper transfer functions satisfying $C(0) = \mathbb{I}_n$ and zero initialization for their state-space realizations.

To sum up, the block diagram shown in Figure 4.4 illustrates the architecture of $\mathcal{L}_1$ adaptive control for the class of systems given by (4.18).

4.4 Contribution 3: L1 adaptive control with nominal FF

The $\mathcal{L}_1$ adaptive control law in (4.20) is independent of any knowledge of the system dynamics in (4.18). For parallel manipulators whose dynamic model is not available or is complex to obtain, the adaptive control law in (4.20) would be very efficient compared to existing non-model-based strategies. Indeed, the $\mathcal{L}_1$ control law considers all the uncertainties as a general unknown nonlinear function with unknown structure. As a matter of fact, $\mathcal{L}_1$ adaptive control inherits the non-model-based adaptive abilities of MRAC while

preserving the robustness of the system. However, it would be useful to include the knowledge that we possess regarding the dynamics of the parallel manipulator in the controller in order to enhance its performance. Especially, the addition of more knowledge about the dynamics improves the tracking performance and avoids high-frequency and high-gain control effort.

4.4.1 Proposed control law

To take advantage of the modeled dynamics of the manipulator, consider the following model-based control law [Bennehar et al., 2015c]

$$\Gamma(t) = A_m r(t) + \Gamma_{ff}(t) + \Gamma_{ad}(t) \tag{4.30}$$

where $A_m \in \mathbb{R}^{n\times n}$ is a *Huruwizt* matrix characterizing the desired transient performance and $\Gamma_{ff}(t) \triangleq M(q_d)\ddot{q}_d + C(q_d,\dot{q}_d)\dot{q}_d + G(q_d)$ is a feedforward term based on the nominal dynamics of the manipulator and the desired trajectories.

The proposed control law in (4.30) can be thought of as the combination of three distinct terms. Each term has a specific role in the closed-loop system.

- The first term $A_m r(t)$ consists of a stabilizing state-feedback term.
- The second term $\Gamma_{ff}(t)$ is a nominal inverse-dynamics-based feedforward term intended to minimize the effect of the inherent nonlinearities of the manipulator.
- The last term $\Gamma_{ad}(t)$, which will be detailed in the subsequent development, is the adaptive signal responsible of rejecting all the remaining disturbances such as friction effects, external disturbances, etc.

The additional model-based feedforward $\Gamma_{ff}(t)$ is in charge of compensation of part of the nonlinear dynamics. In other words, if the modeled dynamics in the feedforward term matches the dynamics of the real system, then the nonlinearities inherent to the system will be reduced.

Substituting the control law (4.30) into (4.18) results in the following error dynamics

$$\dot{r}(t) = A_m r(t) + \Gamma_{ad}(t) - \eta(t,\zeta(t)), \quad r(0) = r_0 \tag{4.31}$$

where $\eta(t,\zeta(t))$, $\zeta = [r^T, q^T]$ is a nonlinear function that gathers all the remaining nonlinearities that result from applying the control law (4.30) to the dynamics of the parallel manipulator (4.18).

Under Assumptions 8, 9 and 10, which remain valid despite the preoposed augmentation, the unknown nonlinear function $\eta(t,\zeta(t))$ can be parametrized using $\|r_t\|_{\mathcal{L}_\infty}$ as follows

$$\eta(t,\zeta(t)) = \theta(t)\|r_t\|_{\mathcal{L}_\infty} + \sigma(t) \tag{4.32}$$

where $\theta(t)$ and $\sigma(t)$ are unknown continuous, piecewise-differentiable and uniformly bounded functions that have to be estimated.

Notice that since the nonlinearity $\eta(t,\zeta(t))$ resulting from the application of the proposed control law in (4.30) is different from the one obtained with the non augmented $\mathcal{L}_1$ adaptive controller in (4.20).

To achieve the desired closed-loop performance, the adaptive control term $\Gamma_{ad}(t)$ should cancel the nonlinear function $\eta(t,\zeta(t))$. However, since this latter is unknown, the controller should estimate it in real-time by means of an adequate adaptation mechanism. According to the $\mathcal{L}_1$ adaptive control theory, the adaptive control component of the control law in (4.30) should be low-pass filtered in order to avoid high-frequency components that may harm the robustness of the controller. Hence, the adaptive control term $\Gamma_{ad}(t)$ is given by its following *Laplace* transform

$$\Gamma_{ad}(s) = C(s)\hat{\eta}(s) \tag{4.33}$$

where $\hat{\eta}(s)$ is the *Laplace* transform of $\hat{\eta}(t)$ and $C(s)$ is a diagonal matrix of filters whose elements are bounded-input bounded-output stable strictly proper transfer functions satisfying $C(0) = \mathbb{I}_n$ and zero initialization for their state-space realizations.

4.4.2 State predictor

To predict the evolution of the combined tracking error, the following state predictor, which is identical to the previously described on in standard $\mathcal{L}_1$ adaptive control, is introduced

$$\dot{\hat{r}}(t) = A_m\hat{r}(t) + \Gamma_{ad}(t) - (\hat{\theta}(t)\|r_t\|_{\mathcal{L}_\infty} + \hat{\sigma}(t)) - Z\tilde{r}(t), \quad \hat{r}(0) = r_0 \tag{4.34}$$

The state predictor in (4.34) mimics the error dynamics in with only two modifications. First, the unknown functions $\theta(t)$ and $\sigma(t)$ are replaced by their estimates $\hat{\theta}(t)$ and $\hat{\sigma}(t)$, respectively. Then, the error rejection term $Z\tilde{r}(t)$ is added to the equation of the predictor.

4.4.3 Adaptation laws

The adaptation estimates are identical to the case of the non-augmented $\mathcal{L}_1$ adaptive controller, they are given by

$$\dot{\hat{\theta}}(t) = \Xi\,\mathrm{Proj}\,(\hat{\theta}(t), P\tilde{r}(t)\|r_t\|_{\mathcal{L}_\infty}), \qquad \hat{\theta}(0) = \hat{\theta}_0 \tag{4.35}$$

$$\dot{\hat{\sigma}}(t) = \Xi\,\mathrm{Proj}\,(\hat{\sigma}(t), P\tilde{r}(t)), \qquad \hat{\sigma}(0) = \hat{\sigma}_0 \tag{4.36}$$

where $\Xi > 0$ is the adaptation gain, $P = P^T > 0$ is the solution to the algebraic *Lyapunov* equation $A_m^T P + PA_m = -Q$ for some arbitrary matrix $Q = Q^T > 0$.

Notice that the nonlinearities to be estimated with the current controller are different from those of the non-augmented one. In other words, since part of the nonlinear dynamics have already been accounted for by the added model-based feedforward term, the remaining non-compensated nonlinearities to be estimated by the adaptive control term

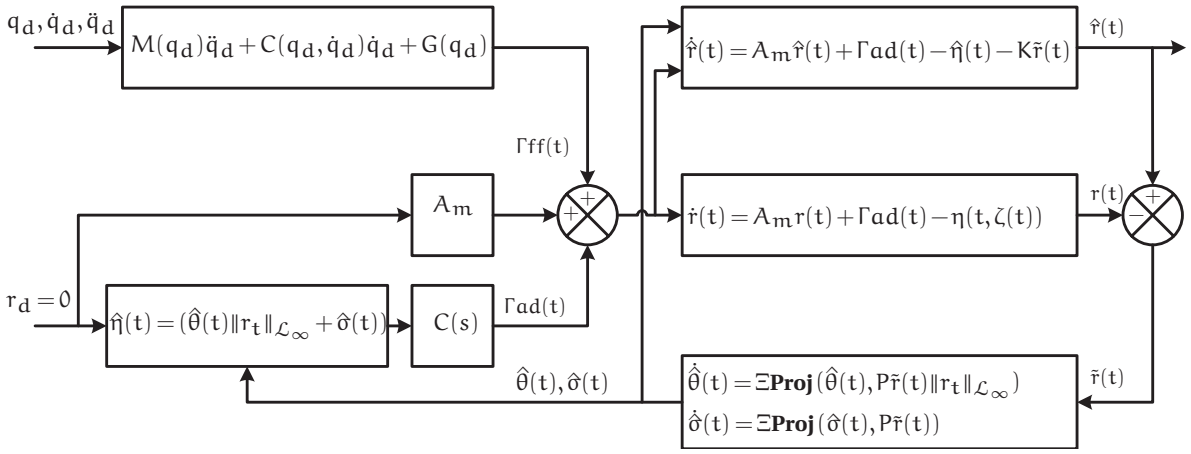


Figure 4.5 – Block diagram of augmented $\mathcal{L}_1$ adaptive control with feedforward dynamics

$\Gamma_{ad}(t)$ are of smaller magnitudes. However, if the feedforward term do not match the real dynamics of the system (wrong parameters or non-accurate modeling), it will be considered as an additional disturbance. Consequently, the adaptive control term Γ_{ad} will have to compensate for more uncertainties which would result in a poorer performance.

To conclude the current subsection, a block diagram explaining the proposed control solution is depicted in Figure 4.5.

4.5 Contribution 4: L1 adaptive control with adaptive FF

In the particular case where the manipulator's dynamics matches the modeled one, the proposed controller in (4.30) would be able to provide better performance than standard $\mathcal{L}_1$ adaptive control. Indeed, since the controller is in charge of rejecting all the uncertainties gathered in the single nonlinear function $\eta(t, \zeta(t))$, the additional feedforward term assists the adaptive control term in canceling the model nonlinearities. However, in practice, the dynamics of the manipulator are very likely to vary or maybe uncertain or unknown. Consequently, including a nominal feedforward that does not match the dynamics of the manipulator might be seen from the control point of view as an additional source of disturbances. Therefore, the proposed controller in (4.30) should consider the case where the dynamics are uncertain, unknown or time-varying.

4.5.1 Proposed control law

The proposed control law that considers the uncertainties of the modeled dynamics is based on a feedforward term with adaptive estimated parameters. It is hence given by

[Bennehar et al., 2015a]:

$$\Gamma(t) = A_m r(t) + Y(q_d, \dot{q}_d, \ddot{q}_d)\hat{\theta}(t) + \Gamma_{ad}(t) \tag{4.37}$$

where $A_m \in \mathbb{R}^{n\times n}$ is a *Hurwitz* matrix that characterizes the desired tracking performance and $\Gamma_{ad}(t)$ is the adaptive control term typical in $\mathcal{L}_1$ adaptive control theory. The novelty of the new proposed control law in is the introduction of the adaptive computed feedforward term $Y(q_d, \dot{q}_d, \ddot{q}_d)\hat{\theta}(t)$ which can be expressed as

$$Y(q_d, \dot{q}_d, \ddot{q}_d)\hat{\theta}(t) = \hat{M}(q_d)\ddot{q}_d + \hat{C}(q_d, \dot{q}_d)\dot{q}_d + \hat{G}(q_d) \tag{4.38}$$

where $\hat{M}(q_d), \hat{C}(q_d, \dot{q}_d)$ and $\hat{G}(q_d)$ are estimates of $M(q_d), C(q_d, \dot{q}_d)$ and $G(q_d)$ respectively. The additional adaptive feedforward term has the ability of adjusting its parameters in real-time according to the variations of the dynamics of the manipulator. Furthermore, if the dynamic parameters of the parallel manipulator are completely unknown, the vector of estimated parameters $\hat{\theta}(t)$ can be initialized to zero. Then, the controller estimates their real values in real-time in order to achieve the desired tracking performance.

Without loss of generality, the application of the proposed control law in (4.5.1) to the dynamics of the parallel manipulator in (4.18) results in error dynamics that can be put in the same form as in the standard $\mathcal{L}_1$ adaptive control which is given by

$$\dot{r}(t) = A_m r(t) + \Gamma_{ad}(t) - \eta(t, \zeta(t)), \quad r(0) = r_0 \tag{4.39}$$

where the remaining dynamics resulting from the feedforward compensation are merged in the unknown nonlinear function $\eta(t, \zeta(t))$. The same assumptions as in standard $\mathcal{L}_1$ adaptive control (Assumptions 8, 9 and 10) are made in order to obtain the following formulation for $\eta(t, \zeta(t))$

$$\eta(t, \zeta(t)) = \theta(t)\|r_\tau\|_{\mathcal{L}_\infty} + \sigma(t) \tag{4.40}$$

Again, to obtain the desired behavior of the tracking error system, characterized by A_m, the adaptive control component $\Gamma_{ad}(t)$ is in charge of canceling the unknown nonlinear function $\eta(t, \zeta(t))$. If the adaptive control term $\Gamma_{ad}(t)$ is identical to the nonlinearity $\eta(t, \zeta(t))$, then the following desired behavior of the tracking error is obtained

$$\dot{r}(t) = A_m r(t) \tag{4.41}$$

However, since $\eta(t, \zeta(t))$ is unknown, the control term $\Gamma_{ad}(t)$ is based on the adaptive estimate $\hat{\eta}(t, \zeta(t))$. Nevertheless, $\hat{\eta}(t, \zeta(t))$ may include undesired high-frequency contents that deteriorates the robustness of the closed-loop system. The $\mathcal{L}_1$ adaptive control formalism consists of filtering these high-frequency harmonics by means of a low-pass filter. Hence, the adaptive control term $\Gamma_{ad}(t)$ in (4.5.1) is given by its *Laplace* transform $\Gamma_{ad}(s)$ as follows

$$\Gamma_{ad}(s) = C(s)\hat{\eta}(s) \tag{4.42}$$

where $\hat{\eta}(s)$ is the Laplace transform of $\hat{\eta}(t) = (\hat{\theta}(t)\|r_t\|_{\mathcal{L}_\infty} + \hat{\sigma}(t))$ and $C(s)$ is a diagonal matrix of filters whose elements are bounded-input bounded-output stable strictly proper transfer functions satisfying $C(0) = \mathbb{1}_n$ and zero initialization for their state-space realizations.

4.5.2 State predictor

The state predictor, required for the evolution of the adaptive estimates is obtained by replacing the unknown linear function $\eta(t,\zeta)$ in (4.39) by its online estimate $\hat{\eta}(t,\zeta)$. It is written as

$$\dot{\hat{r}}(t) = A_m\hat{r}(t) + \Gamma_{ad}(t) - (\hat{\theta}(t)\|r_\Gamma\|_{\mathcal{L}_\infty} + \hat{\sigma}(t)) - K\tilde{r}(t), \quad \hat{r}(0) = r_0 \tag{4.43}$$

The obtained prediction error issued from the predictor will be used to drive the adaptation laws in order to estimate the uncertainties.

4.5.3 Adaptation laws

The proposed control law in (4.5.1) contains estimates of unknown elements whose evolution with respect to time should be governed by an adequate adaptation laws. The controller in (4.5.1) involves three terms to be estimated; namely, the unknown parameters vector $\hat{\Theta}(t)$ and the two unknown nonlinear functions $\hat{\theta}(t)$ and $\hat{\sigma}(t)$ commonly seen in

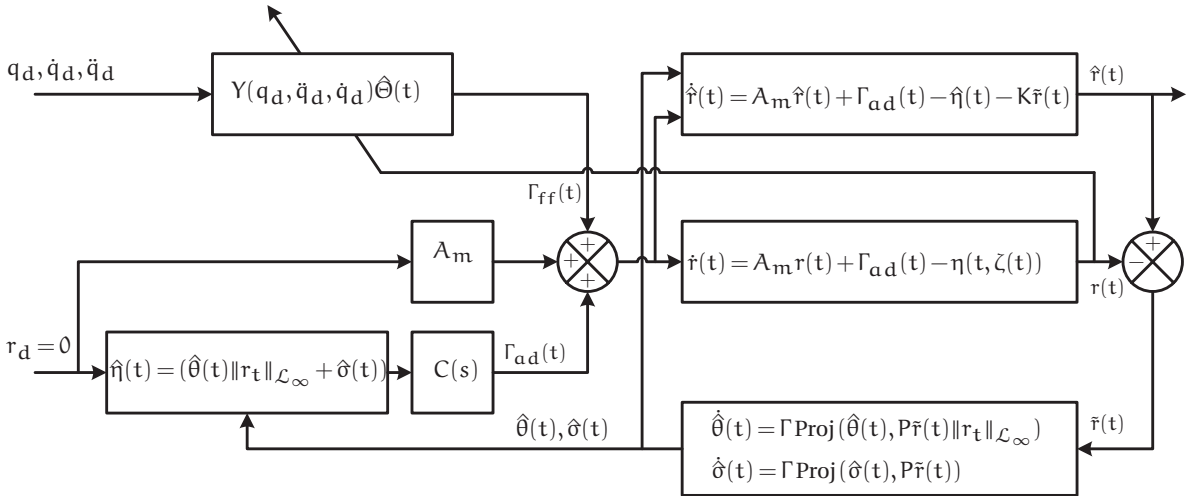


Figure 4.6 – Block diagram of $\mathcal{L}_1$ adaptive control with adaptive feedforward

standard $\mathcal{L}_1$ adaptive control. The following adaptation laws are introduced

$$\dot{\hat{\theta}}(t) = \Xi \text{Proj}\left(\hat{\theta}(t), P\tilde{r}(t)\|r_t\|_{\mathcal{L}_\infty}\right), \qquad \hat{\theta}(0) = \hat{\theta}_0 \tag{4.44}$$

$$\dot{\hat{\sigma}}(t) = \Xi \text{Proj}\left(\hat{\sigma}(t), P\tilde{r}(t)\right), \qquad \hat{\sigma}(0) = \hat{\sigma}_0 \tag{4.45}$$

$$\dot{\hat{\Phi}}(t) = -KY(q_d, \dot{q}_d, \ddot{q}_d)^T r(t), \qquad \hat{\Phi}(t) = \hat{\Phi}_0 \tag{4.46}$$

It can be seen that the prediction error is used to adjust the estimates of $\theta(t)$ and $\sigma(t)$, while the combined error $r(t)$ is used to adjust the parameters of the adaptive feedforward. It should be noticed that the regression matrix $Y(.)$ in adaptation law is evaluated along the desired trajectories $q_d, \dot{q}_d$ and $\ddot{q}_d$. This means that its values can be computed offline and then fed to the controller, which reduces the computation cost of the controller.

The block diagram of the proposed controller is shown in Figure (4.6)

4.6 How to deal with the issue of internal forces ?

Parallel manipulators with actuation redundancy have various advantages compared to their non-redundant homologous. Their most noteworthy ones include higher accuracy, improved stiffness and more dexterity. Hence, actuation redundancy promotes parallel manipulators to be perfect candidates for high-demanding applications such as robot-assisted surgery, electronic devices assembly and micro-manipulation tasks. However, redundantly actuated PKMs exhibit undesired features that should be carefully addressed. One of the main disadvantages of actuation redundancy that may deteriorate the performance of the manipulator is the presence of internal forces in the mechanism. These forces are mainly due to conflicting actuators' efforts since there is more inputs than required to accomplish a specific end-effector motion.

In the sequel, we briefly enumerate the main sources of antagonistic internal forces in redundantly actuated parallel manipulators and their negative effect on the overall performance. Then, we describe the adopted solution to solve this issue in our proposed controllers.

4.6.1 Sources of internal forces

According to [Muller, 2009], antagonistic control forces are mainly due to the following sources:

1. Uncertainties in the model geometry.
2. Measurement errors.
3. Non-synchronized control of the actuators.

While the uncertainties in the model geometry can be addressed using identification techniques, measurement errors are directly related to the instrumentation of the manipulator and should be explicitly considered in the control design. The last item is straightly related to the control strategy itself. Though non-synchronized control might seem only related to decentralized controllers, using model-based controllers could lead to internal forces as well.

4.6.2 Their effects

The effect of geometric uncertainties on two of the best known model-based control schemes; namely, the augmented PD and the computed torque controllers, was addressed in [Muller, 2010]. The application of the unaltered versions of these controllers on an uncertain redundantly actuated parallel manipulators revealed that the way in which the control inputs act on the system is changed due to geometric uncertainties.

The effect of measurement errors on both decentralized and model-based control schemes was addressed in [Muller and Hufnagel, 2011]. It was shown that measurement errors degrade the performance of both decentralized and model-based controllers and, hence, worsen the accuracy of the control system and increases the energy consumption. Furthermore, it was demonstrated that decentralized control inherently enables antagonistic control forces that are not compatible with the kinematics of the redundantly actuated parallel manipulator. Hence, decentralized controllers cannot be straightforwardly applied to redundantly actuated parallel manipulators without resolving such incompatibility.

For all of the above reasons, the actuation redundancy of the manipulator has to be addressed in the control design before applying the computed control inputs to the manipulator.

4.6.3 Adopted solution

Antagonistic internal forces may be used for secondary tasks such as backlash avoidance [Muller, 2005] and stiffness modulation [Muller, 2006]. However, if no secondary task is scheduled within the control scheme, one should consider reducing the effects of the internal forces in order to avoid vibrations that might deteriorate the control performance and increase the energy consumption.

As it has been explained in [Muller and Hufnagel, 2011], the internal forces are caused by the control inputs which are in the null space of the inverse Jacobian matrix J_m. Consequently, one solution to reduce their effects, can be to project the generated control inputs into the range space of J_m using the following projector

$$R_{J_m^T} = (J_m^+)^T J_m^T \tag{4.47}$$

Hence, the actual control inputs that will be applied on the robot are the "regularized" ones obtained through the following projection

$$\Gamma^*(t) = R_{J_m^T}\,\Gamma(t) \tag{4.48}$$

In this thesis, we will be dealing with different types of parallel manipulators belonging to both classes; redundantly and non-redundantly actuated. Consequently, if the parallel manipulator is with actuation redundancy, the control inputs are first projected using (4.48) before being applied to the manipulator. Otherwise, if the parallel manipulator is not redundantly actuated, the regularization (4.48) will have no effect on the control inputs. Indeed, in the case of non-redundantly actuated manipulators, the inverse Jacobian matrix is squared and, hence, is invertible (away from singularities). Consequently, the projection matrix $R_{J_m^T}$ will be equal to the identity matrix.

4.7 Conclusion

In this chapter, the application of $\mathcal{L}_1$ adaptive control on parallel manipulators is presented. To better explain the motivations behind the development of this novel control strategy, we first considered a first-order single-input single-output system and explained how the passing from Model Reference Adaptive Control to $\mathcal{L}_1$ adaptive control is performed. After that, its extension to parallel manipulators is described and the various blocks belonging to its architecture are introduced.

Since $\mathcal{L}_1$ adaptive control does not depend on any knowledge of the dynamics of the system, its performance can be improved by including the dynamic model of the parallel manipulator in the control loop. In this context, we first proposed to include a nominal feedforward based on the inverse dynamics and the desired trajectories. Then, we further extend this idea to handle the case where the modeled dynamics are unknown, uncertain or time-varying. Towards that goal, we augmented the architecture of $\mathcal{L}_1$ adaptive control with an adaptive inverse dynamics-based feedforward.

In the next chapter, we will present our experimental platforms that will serve as real-time validation testbeds for our proposed control solutions. Then, each control strategy is validated on a real parallel manipulator.

Part III

Real-time experiments and results

CHAPTER

5 Experimental platforms and implementation issues

Contents

5.1 Introduction

This chapter is devoted to the description of the experimental platforms used during this thesis for the real-time validation of the proposed control solutions. They are divided into two categories according to their actuation type; namely, redundantly and non-redundantly actuated. A description of each parallel manipulator, its parameters and its development system used to implement the control scheme are provided. Moreover, the adopted strategy for generating the reference trajectories, required in some controllers, is also described. Finally, the performance criteria that will be used to evaluate and compare the performance of each controller based on its tracking errors is explained.

5.2 Description of non-redundant platforms

5.2.1 Delta: a 3-DOF PKM

The Delta robot is a 3-DOF parallel manipulator developed at EPFL in Lausanne, Switzerland. It consists of three identical kinematic chains that link the end-effector (traveling-plate or nacelle) to the fixed-base. Each kinematic chain is a serial succession of an actuator, a rear-arm and a forearm.

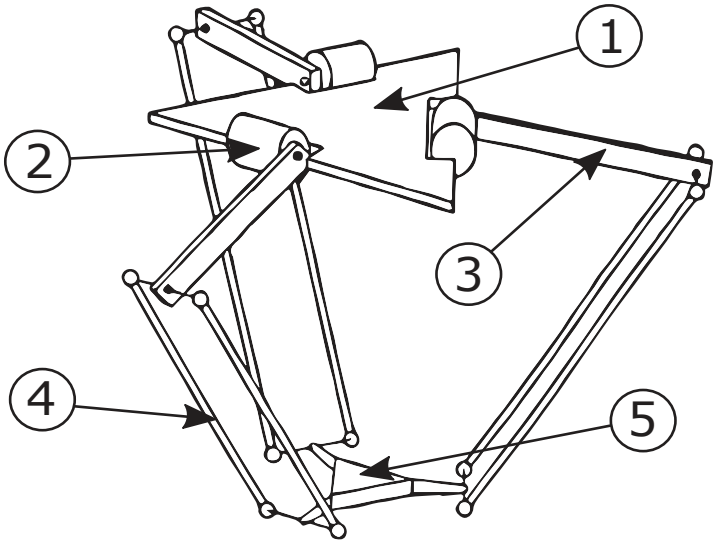


Figure 5.1 – CAD of the Delta robot; 1: Fixed base, 2: actuator, 3: rear-arm, 4:forearm, 5: traveling plate.

The mechanical structure is actuated through three direct-drive actuators mounted on the fixed-based providing each one 20 Nm of maximum torque. The motion of the traveling-plate is achieved by transmitting the rotations of the actuators to the rear-arms, which themselves are transmitted to the forearms which are linked at the traveling-plate

Table 5.1 – Main parameters of the Delta robot

Parameter	**Value**	**Parameter**	**Value**
Rear-arm's length	240 mm	Rear-arm's mass	0.22 kg
Forearm's length	480 mm	Forearm's mass	0.084 kg
Actuator's inertia	1.82 10^{-3} kg.m^2	Traveling-plate's mass	0.305 kg

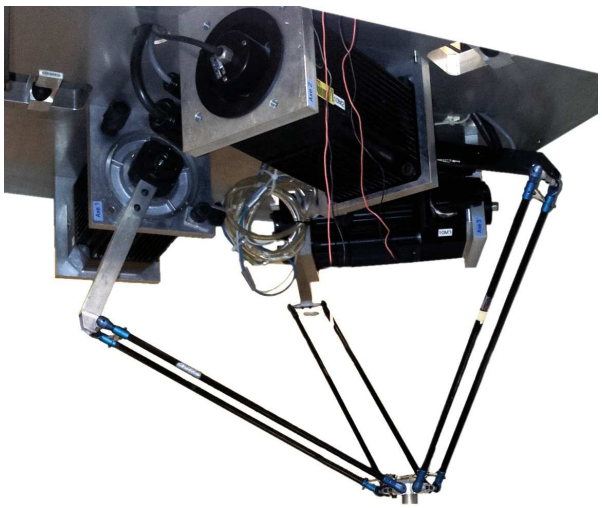


Figure 5.2 – The fabricated Delta robot

through pairs of ball-and-socket passive joints. The particular use of parallel rods for the forearms allows to restrict the motion of the traveling plate to three translations along x, y and z-axes. Figure 5.1 shows a schematic design of a typical Delta robot.

The set of geometric and dynamic parameters of the Delta robot used for experiments in this thesis is summarized in Table 5.1.

The development of the control scheme on the EPFL's Delta robot is done with C language via Visual Studio software from Microsoft. The controller is run under a frequency of 1 KHz (sampling time of 1 ms). Figure 5.2 shows the fabricated 3-DOF Delta robot which is located in the Robotic Systems Laboratory of EPFL. This robot was proposed to validate the adaptive RISE controller within a collaboration with EPFL.

5.2.2 VELOCE: a 4-DOF PKM

The VELOCE robot is a 4-DOF parallel manipulator designed and fabricated in LIRMM. It is a fully parallel manipulator, which means that it has the same number of kinematic

Table 5.2 – Main parameters of VELOCE robot

Parameter	Value	Parameter	Value
Rear arms length	0.2 m	Rear arms mass	0.541 kg
Forearms length	0.53 m	Forearms mass	0.08 kg
Actuators' inertia	0.0041 kg.m^2	Traveling plate's mass	0.999 kg

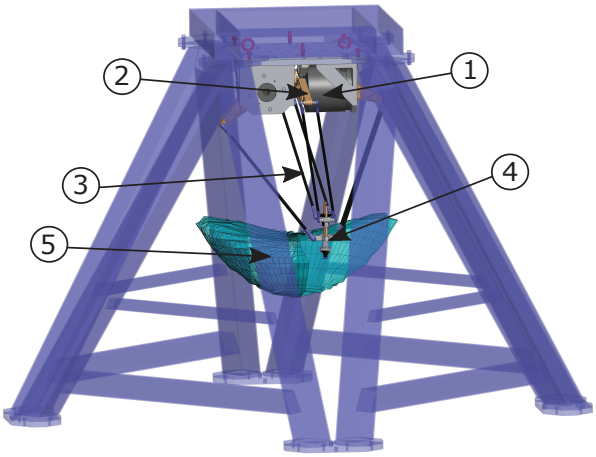


Figure 5.3 – CAD of the VELOCE robot, 1: actuator, 2: rear-arm, 3: forearm, 4: traveling plate, 5: workspace.

chains as DOFs [Merlet, 2006]. Each kinematic chain can be seen as a serial arrangement of an actuator (i.e. its stator part), an arm (including the rotor part of the actuator) and a forearm. Its end-effector (a traveling plate) is able to perform three spatial translations x, y, and z, and one rotation α about the vertical axis perpendicular to its fixed-base. The CAD design of VELOCE as well as its workspace are illustrated in Fig. 5.3. Table 5.2 summarizes the main geometric and dynamics parameters of the VELOCE robot.

The actuators of fabricated VELOCE robot are the TMB0140-100-3RBS ETEL direct-drive motors. They can provide a maximum peak torque of 127 Nm and are able to reach 550 rpm of speed. Each actuator is equipped with a non-contact incremental optical encoder providing a total number of 5000 pulses per revolution. The global structure of the manipulator is able to reach 10 m/s of maximum velocity of the traveling plate, 200 m/s^2 of maximum acceleration and is able to handle a maximum payload of 10 Kg. The control architecture of the VELOCE robot is implemented using *Simulink* from from *Mathworks Inc.* and compiled using *XPC Target* and the *Real-Time* toolboxes. The resulting low-level code is then uploaded to the target PC; an industrial computer cadenced at 10 KHz (i.e. sample time of 0.1 ms). The experimental testbed is displayed in Fig. 5.4.

The VELOCE robot will be used as an experimental testbed for the validation of the $\mathcal{L}_1$ adaptive control theory (Contribution 3) and its augmented version with nominal feedforward (Contribution 4).

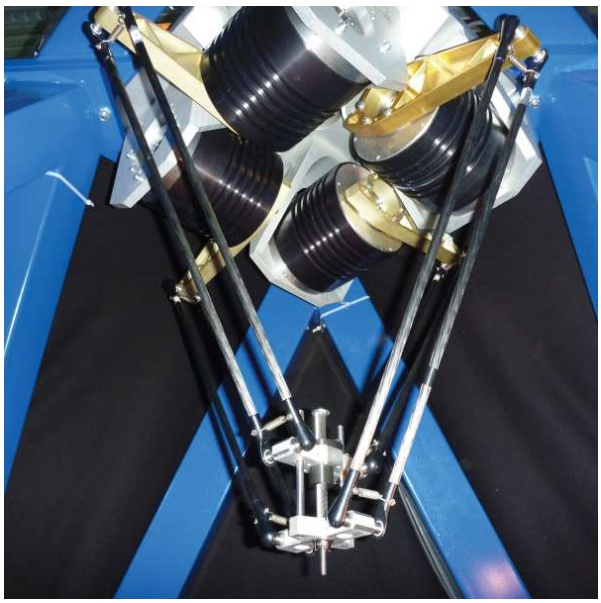


Figure 5.4 – The fabricated VELOCE robot

5.3 Description of redundant platforms

5.3.1 DUAL-V: a 3-DOF RA-PKM

The first parallel manipulator ever used for experimental validation of the developed control schemes within this thesis was the DUAL-V robot. This prototype has been devel-

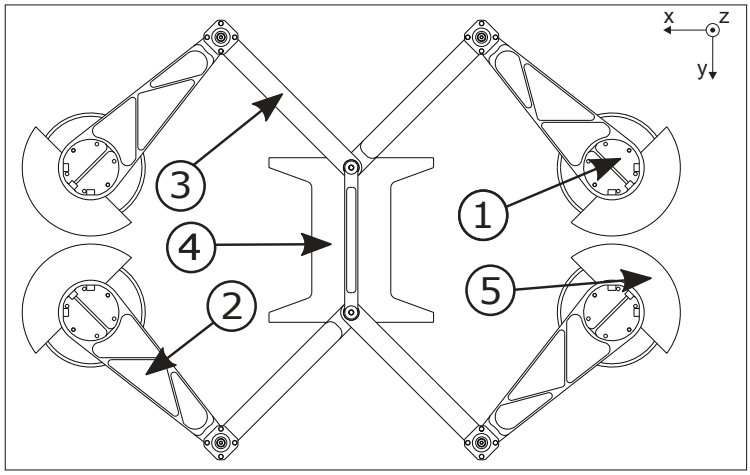


Figure 5.5 – Illustration of the CAD model of Dual-V robot. 1: actuator, 2: rear-arm (crank), 3: forearm (coupler), 4: traveling plate and 5: counter-mass.

Parameter	Value	Parameter	Value
Cranks' length	0.28 m	Coupler' length	0.28 m
Crank's mass	1.169 kg	Coupler' mass	0.606 kg
Cranks' inertia	0.01297 kgm^2	Couplers' inertia	0.00642 kgm^2
Platform's length	0.22 m	Counter-masses inertia	0.02685 kgm^2
Platform's mass	0.899 kg	Counter-masses mass	7.983 kg
Platform's inertia	0.22 kgm^2	Actuator's inertia	0.0041 kgm^2

Table 5.3 – Main parameters of DUAL-V robot

oped in the framework of a collaboration between LIRMM and the Laboratory of Mechanical Automation and Mechatronics of the University of Twente in the Netherlands. The DUAL-V robot was mainly designed to be a dynamically balanced parallel manipulator, meaning that inertial forces and moments exerted by the manipulator on its base are minimized by design. These forces and moments, called shaking forces and shaking moments, are a significant source of vibrations of the robot's base. In addition to control, vibrations of neighboring machines are an important cause of wear and failure of the manipulator [van der Wijk et al., 2013].

To achieve the dynamic balancing, additional counter-masses are added to the manipulator's structure. These counter-masses enable similar and opposite motions of the different elements. As explained in [van der Wijk et al., 2013], the more the motions are similar and opposite, the better the balancing of the structure is. Experimental comparison between the balanced manipulator and the unbalanced one (i.e. without the balancing counter-masses) show that the DUAL-V has up to 97 % lower shaking forces and up to 96 % lower shaking moments.

DUAL-V is a planar redundantly actuated parallel manipulator with three DOFs. Its architecture, depicted in Figure 5.5, consists of four kinematic chains connecting a common moving platform (via the C_i points) to the fixed base through four actuated pivots (via the A_i points). Each kinematic chain is a serial arrangement of actuator's stator, a crank (with the actuator's rotor) and a coupler rod. The moving platform can move along the horizontal plane defined by x and y axes, and perform one rotations around the axis perpendicular to the base.

The main geometric and dynamic parameters of the different parts of the DUAL-V robot are summarized in Table. 5.3. The inertia values are given with respect to the axis of rotation of the specified element.

The fabricated prototype of the DUAL-V robot is depicted in Figure. 5.6. The manipulator's kinematic chains are made of aluminum, while the counter-masses are made of brass. The manipulator is actuated through four *ETEL RTMB0140-100* direct drive actua-

tors mounted fixed base of $1 \times .8 \times .025$ m made with aluminum as well. Each actuator is capable of providing a maximum torque of 127 Nm.

The experimental setup of the DUAL-V robot is shown in Figure 5.7. The control development is implemented using *Simulink* software from *Mathworks Inc.* The *Real-Time Workshop* toolbox, also from *Mathworks Inc.*, is used to compile the designed control scheme and obtain low level code. The latter is uploaded to the industrial *Target PC* which runs it in real-time, communicates with the drives and provides the measured information from the sensors at the same frequency of 10 kHz.

The DUAL-V robot was mainly used to validate the proposed control schemes developed in the context of solution 1; namely, the extended DCAL and the adaptive RISE controllers. For more details about the theoretical development behind the design of the DUAL-V robot, the reader is referred to [van der Wijk et al., 2013].

5.3.2 ARROW: a 6-DOF RA-PKM

As it was described previously, the ARROW robot is the redundantly actuated parallel manipulator prototype developed at LIRMM within the ARROW project. Along with its 1-DOF rotational turntable, the entire structure has five DOFs. The parallel module of the ARROW robot consists of six linear actuators linked to a moving platform having four DOFs. Thus, the allowed motion for the moving platform is of 3T-1R type; three spatial translations and one rotation along the vertical axis parallel to the fixed-base.

The parallel module has two degrees of actuation redundancy while the turntable has one. Consequently, the entire machine has three degrees of actuation redundancy; eight actuators (six of them are linear while the remaining two are rotational). From a control point of view, the most relevant part to be controlled is the parallel module. Besides, the

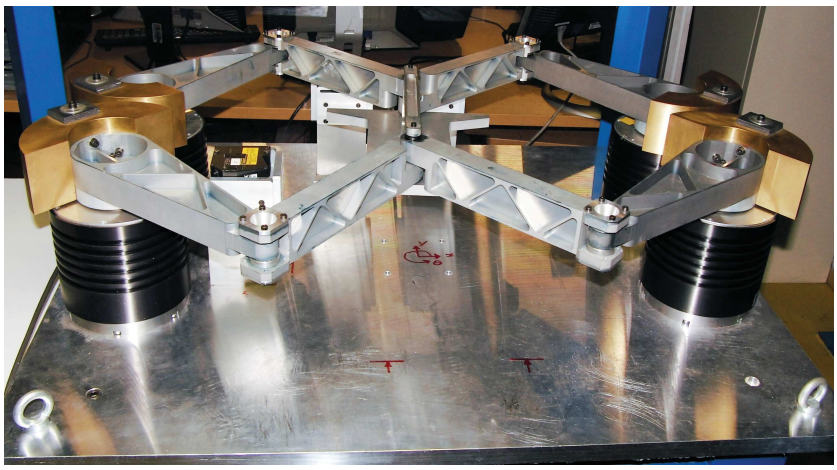


Figure 5.6 – The fabricated DUAL-V robot

Figure 5.7 – Experimental setup of the DUAL-V robot

allocated time for this doctoral thesis didn't permit to test the entire machine. Therefore, we focus our following discussion only on the parallel module.

The six linear actuators of the parallel module are aligned along the same direction and arranged into two sets. Each set gathers three actuators lying in the same plane and equidistant one from another. The symmetrical structure of the robot is highly favorable

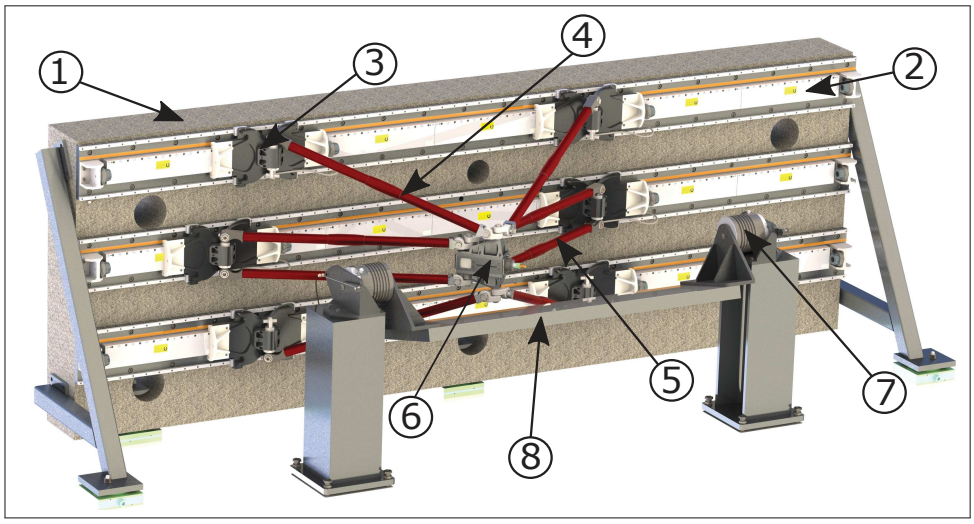


Figure 5.8 – CAD of the ARROW robot, 1: fixed-base, 2: slider, 3: linear actuator, 4: simple arm, 5: parallelogram, 6: moving platform, 7: turntable's actuator, 8: turntable.

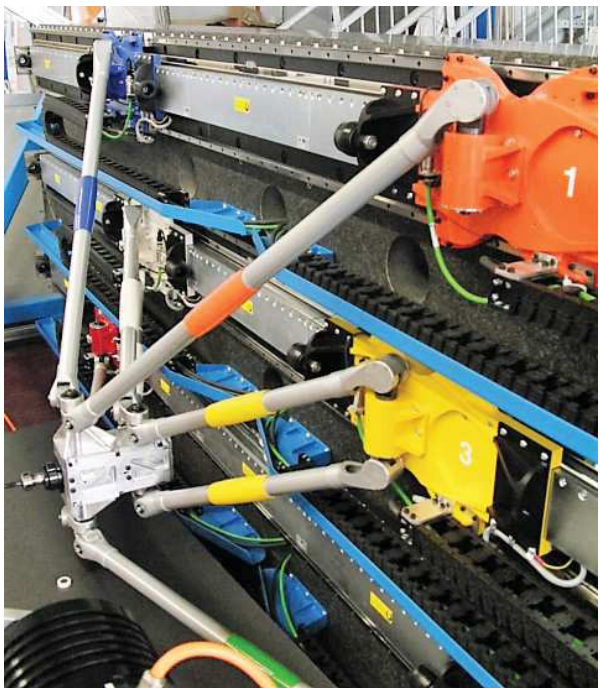


Figure 5.9 – The fabricated ARROW robot

from the manufacturing point of view. In addition, the arrangement of the actuators enables a large workspace along the axis they are aligned along. Moreover, the moving platform of the ARROW robot is able to rotate with ±90 degrees. The principle of functioning of the mechanism is quite simple. The pair of parallelograms in the middle constraints the rotations of the moving platform to only one rotation about the vertical axis parallel to the fixed-base. Then, all the chains contribute to positioning and orientating the moving platform.

The set of various geometric and dynamic parameters of the different parts of the ARROW robot are summarized in Table 5.4. The rotational inertia of the platform is given with respect to its rotation axis.

Parameter	**Value**	**Parameter**	**Value**
Simple slider cart's mass	11.1 kg	Parallelogram slider cart's mass	11.34
Simple arm linear mass	1.744 kg/m	Parallelogram arm linear mass	3.488 kg/m
Simple arms' length	0.96 m	Parallelograms arms' length	0.61 m
Platform's mass	10.2 kg	Platform's inertia	0.414 kgm^2

Table 5.4 – Main parameters of ARROW robot

Figure 5.10 – Experimental setup of the ARROW robot

The manufactured ARROW robot is shown in Figure 5.9. Its actuators are all identical of type *Ironless ETEL ILM12-060*, providing each one a maximum force of 2500 N.

The experimental setup of the ARROW robot is displayed in Figure 5.10. In a similar manner to the two previous prototypes, *Simulink* and *Real-Time Workshop* from *Mathworks Inc.* are used to implement the control scheme and to generate the real-time execution code. The only difference is that the target PC is running under 5 kHz of frequency. The main reason for increasing the sampling time is to give more space for the security blocks computations.

5.4 Generation of reference trajectories

In order to implement any control scheme for mechanical manipulators, the desired trajectories to be followed by the end-effector manipulator have to be available for the controller at each sample time. While some control schemes require only the desired position to be available, other schemes may require other higher-order trajectories such as velocity, acceleration, jerk, etc. Since most our developed solutions are in joint-space and rely on the dynamics of the system being fed to the controller as a feedforward control term, desired joint positions, velocities and acceleration trajectories have to be generated at each sample time. Although the subject of trajectory generation exceeds the scope of this thesis, a brief description of the adopted solution in our experiments is provided in the sequel.

Naturally, the desired path of the robot is specified in Operational space through several way points. Then, the corresponding joint-space trajectories are obtained through the adequate kinematic relationships. In this thesis, the reference trajectories to be tracked by the robot belong to the class of point-to-point trajectories [Khalil and Dombre, 2004]. The end-effector of the robot has to move from an initial Cartesian position denoted by X_0 to a

final one denoted by X_f with zero velocity and acceleration. Once the end-effector reaches the final position X_f, the next target position is fed to the trajectory generator. Hence, the previous final position becomes the new X_0, while the new target position becomes X_f, and so on.

The evolution of the trajectory between X_0 and X_f with respect to time is governed by the following equation

$$X_d(t) = X_0 + Dr(t) \tag{5.1}$$
$$\dot{X}_d(t) = D\dot{r}(t) \tag{5.2}$$
$$\ddot{X}_d(t) = D\ddot{r}(t) \tag{5.3}$$

where $D \triangleq X_f - X_0$ is the distance to be run through and $r(t)$ is a fifth degree polynomial given by

$$r(t) = 10\left(\frac{t-t_0}{T}\right)^3 - 15\left(\frac{t-t_0}{T}\right)^4 + 6\left(\frac{t-t_0}{T}\right)^5, \qquad r(t_0) = 0, r(t_0+T) = 1 \tag{5.4}$$

where t_0 is the initial time of the trajectory and T its duration that has to be chosen by the user with careful consideration of the manipulator's capacities.

To compute the joint trajectory $q_d(t)$ corresponding to the Cartesian trajectory $X_d(t)$, the inverse geometric model is required. This is often not a complex task since the inverse geometric model for parallel manipulators is usually easy to obtain, unlike the forward geometric model which often has more than one solution. To obtain the corresponding desired joint velocities and accelerations, two solutions can be used depending on the availability of the Jacobian matrix and its first time derivative.

In the case of absence of the Jacobian matrix J_m and its time derivative $\dot{J}_m$, numerical differentiation is used to obtain $\dot{q}_d(t)$ and $\ddot{q}_d(t)$. However, if the J_m and its first time-derivative $\dot{J}_m$ are known, we use the following formula to obtain the joint velocities and accelerations

$$\dot{q}_d(t) = J_m \dot{X}_d(t) \tag{5.5}$$
$$\ddot{q}_d(t) = J_m \ddot{X}_d(t) + \dot{J}_m \dot{X}_d(t) \tag{5.6}$$

An illustrative example of one dimensional trajectory obtained using this method is illustrated in Figure 5.11.

5.5 Performance evaluation criteria

To evaluate the performance of the proposed controllers, it is necessary to formulate some criteria that allow the quantification of such performance. Since the main objective of the current work is to improve the tracking performance of parallel manipulators, it is

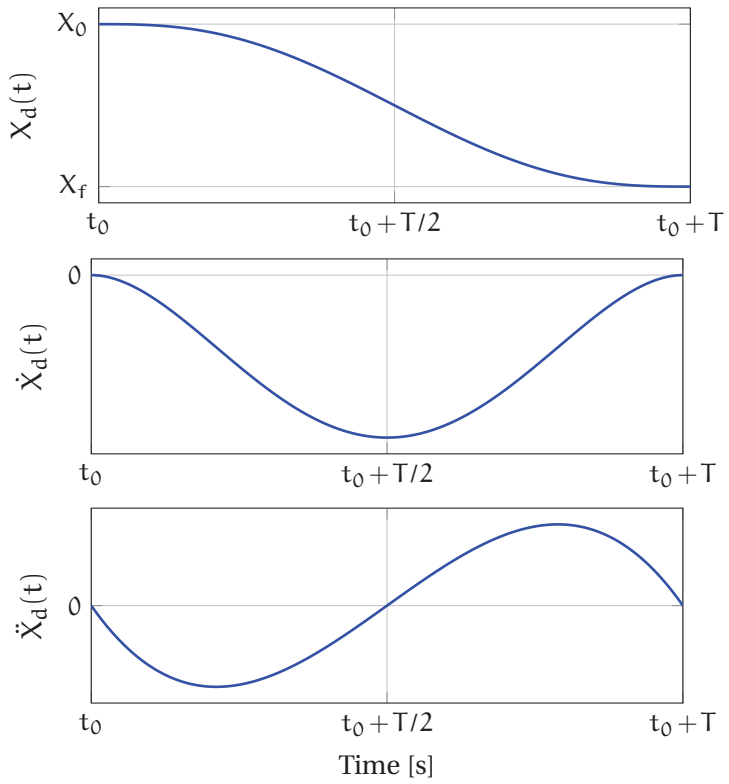


Figure 5.11 – Example of a point-to-point reference trajectory

then straightforward that the adopted criteria would be based on the tracking errors of the reference trajectories. The solution that is often introduced in the literature is the **R**oot **M**ean **S**quare **E**rror (RMSE).

Consider a mechanical manipulator equipped with an end-effector that is capable of performing all available motions in the Cartesian space (three translations and three rotations). The measured Cartesian coordinates of its end-effector at discrete time step k are given by $\hat{X}(k) = [\hat{x}(k), \hat{y}(k), \hat{z}(k), \hat{\alpha}(k), \hat{\beta}(k), \hat{\gamma}(k)]^T$, while the desired Cartesian position of the end-effector at the same discrete time step k is given by $X_d(k) = [x_d(k), y_d(k), z_d(k), \alpha_d(k), \beta_d(k), \gamma_d(k)]^T$. The tracking error vector at discrete time step k is hence given by $e_c(k) \triangleq X_d(k) - \hat{X}(k) = [e_x(k), e_y(k), e_z(k), e_\alpha(k), e_\beta(k), e_\gamma(k)]$.

Evaluating the measurement error for each time step $k \in [1, N]$, we end up by N-dimensional error vector for each DOF. Consider the following RMSE criteria based on the

computed tracking errors

$$RMSE_T = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(e_x^2(k)+e_y^2(k)+e_z^2(k))} \tag{5.7}$$

$$RMSE_R = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(e_\alpha^2(k)+e_\beta^2(k)+e_\gamma^2(k))} \tag{5.8}$$

Similarly, to evaluate the joints tracking performance, we define the following RMSE criteria based on the joint tracking errors

$$RMSE_J = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(\sum_{l=1}^{n} e_{q_1}^2(k)+e_{q_2}^2(k)+\cdots+e_{q_l}^2(k))} \tag{5.9}$$

where $e_{q_l}(k)$ is the tracking error of joint $l \in [1 \ldots n]$ at time step k. Obviously, not all robotic manipulators have six DOFs. In this case, we only consider the available DOFs when evaluation the RMSE criteria. For instance, in the case of the DUAL-V robot, RMS_T consider only the motion along the x and y axes, while the $RMSE_R$ criterion considers only the rotational DOF α.

5.6 Conclusion

In this chapter we have described the various parallel manipulators that will serve as experimental testbeds to validate our control schemes. Four parallel manipulators were introduced, each one has its own characteristics and typical applications. To show that the proposed control schemes can be applied on both redundant and non-redundant manipulators, we considered two manipulators from each family. In the non-redundant family, we have introduced the Delta 3-DOF robot and the VELOCE robot 4-DOF robot, which is similar to the Delta but with four identical chains instead of three. In the redundant family we have described two prototypes. The first one is the DUAL-V 3-DOF planar parallel manipulator which have the peculiarity of being dynamically balanced. Then, we have presented the prototype developed at LIRMM within the ARROW project. It is a 4-DOF parallel machine intended to be used for industrial machining.

Then we have described the adopted method for the generation of the reference desired trajectories, including positions, velocities and accelerations. These trajectories are required for the implementation of our control schemes and have to be adequately designed. At last, we have introduced some criteria based on the root mean square of the tracking errors to evaluate the performance of each of our controllers. It is necessary to

highlight the benefits of the proposed solutions, especially in terms of tracking performance.

It remains to discuss in the next two chapters the obtained real-time experimental results. Since we have divided our control solutions into two main categories, the experimental discussion will similarly follow the same organization. Hence, in the next chapter we will start by discussing the results of **Solution 1** which includes two proposed controllers.

CHAPTER

6 Experimental validation of solution 1

Contents

6.1 Introduction

In this chapter, experimental results of the two proposed controllers included in **Solution 1** are presented and discussed. Both controllers are validated on the DUAL-V redundantly actuated planar parallel manipulator. For comparison purposes, the extended DCAL controller with nonlinear feedback gains is compared with the standard DCAL controller. From the other hand, the proposed adaptive RISE-based controller is compared with original RISE control. Furthermore, to evaluate the robustness of the proposed control solutions towards payload change, two scenarios are proposed for each controller case. The first scenario is the nominal one where no additional payload is considered. In the second scenario, an additional payload with unknown dynamic properties is added to the traveling-plate of the DUAL-V robot.

6.2 Experimental results of contribution 1 on Dual-V

In the following we will provide and discuss the obtained experimental results of the proposed extended DCAL with nonlinear feedback gains on the DUAL-V robot. To demonstrate the relevance of the proposed solution, both the proposed and standard DCAL are implemented and compared. Two scenarios are experienced to test the robustness of the control scheme and its adaptive algorithm towards payload change. The parameters of both controllers used in both scenarios are summarized in Table 6.1. The DCAL controller parameters were obtained through trial and error technique. Regarding the extended DCAL controller, the tuning algorithm described in 3.2.5 has been used.

Table 6.1 – Parameters of the standard and extended DCAL controllers

Standard DCAL		Extended DCAL	
Parameter	Value	Parameter	Value
		k_p	44×10^3
		δ_1	2×10^{-3}
K_p	35×10^2	α_1	1.45
K_v	25	k_v	75
K	$7.5 \times \text{diag}(1, 10^{-1})$	δ_2	2×10^{-2}
		α_2	1.3
		K	$7.5 \times \text{diag}(1, 10^{-1})$

Fifth-order polynomials, described in 5.4, have been used to generate the desired reference trajectories. The traveling-plate of the DUAL-V has to move throughout its workspace across different configurations, including specific rotations.

6.2.1 Scenario 1: nominal case

In this scenario, no payload was added to the traveling-plate of DUAL-V robot. However, we suppose that its dynamic parameters are unknown. More accurately, we consider that the mass m_{tp} and inertia I_{tp} of the traveling plate should be estimated.

A comparison between the Cartesian tracking errors of both controllers is illustrated in Figure 6.1. The plots are zoomed within the interval [11,14] seconds, for clarity. It can be seen that the proposed extended DCAL controller with nonlinear feedback gains provides better results than the standard DCAL. Thanks to their time-varying property, the nonlinear feedback gains adjust themselves depending on to the magnitudes of the tracking errors. This allows to quickly reject large errors while avoiding large overshoots resulting in improved tracking.

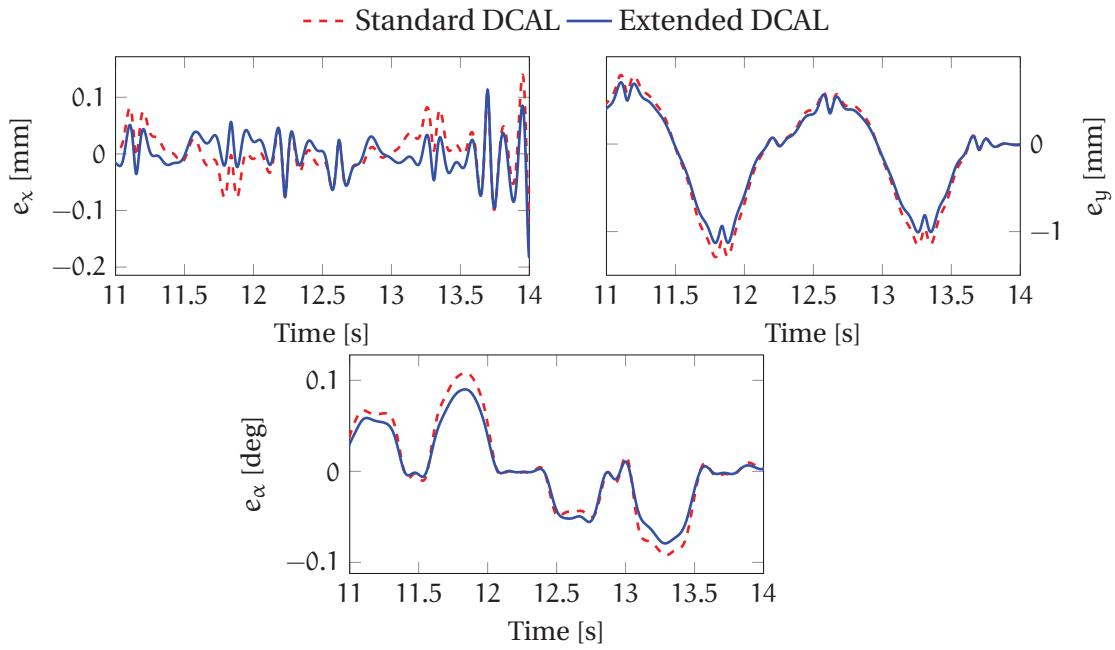


Figure 6.1 – Scenario 1 - Cartesian tracking errors

The previously introduced RMSE criteria are used to evaluate the enhancement brought by the use of the nonlinear feedback gains in the extended controller. The obtained results are summarized in Table 6.2 from which we can see that the tracking errors are reduced by up to 13% compared to the standard controller.

The evolution of the estimated parameters is depicted in Figure 6.2. Notice that these parameters were initialized to zero. Unsurprisingly, the estimated parameters converge to their real values in the case of standard DCAL since this controller is known for its good adaptation performance. The proposed modification for DCAL, however, does not hurt this feature as it can be observed from Figure 6.2. Indeed, the estimated parameters in both controllers converge within the same time span. Furthermore, it should be noticed that the steady state values of the estimated parameters (mass and the inertia of the traveling-plate) match those given by the CAD. It can also be noticed that the inertia parameter I_{tp} does not

Table 6.2 – Scenario 1: Evaluation and comparison of the tracking performance.

	$RMSE_J$ [deg]	$RMSE_T$ [mm]	$RMSE_R$ [deg]
Standard DCAL	0.1279	48.173×10^{-2}	4.01×10^{-2}
Extended DCAL	0.1107	41.901×10^{-2}	3.63×10^{-2}
Improvement	13.45 %	13 %	9.5 %

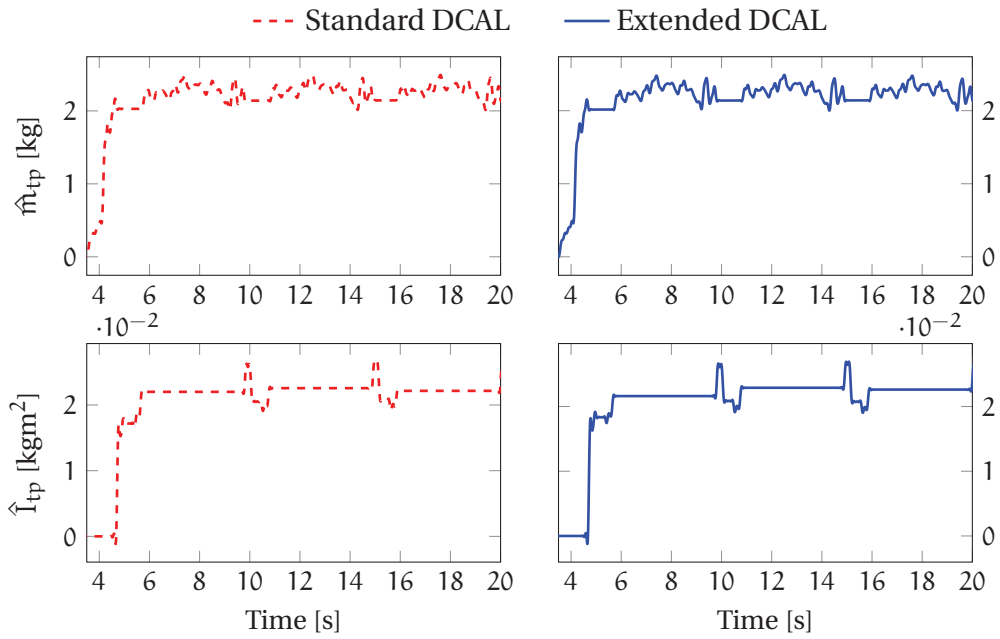


Figure 6.2 – Scenario 1 - Evolution of the estimated parameters

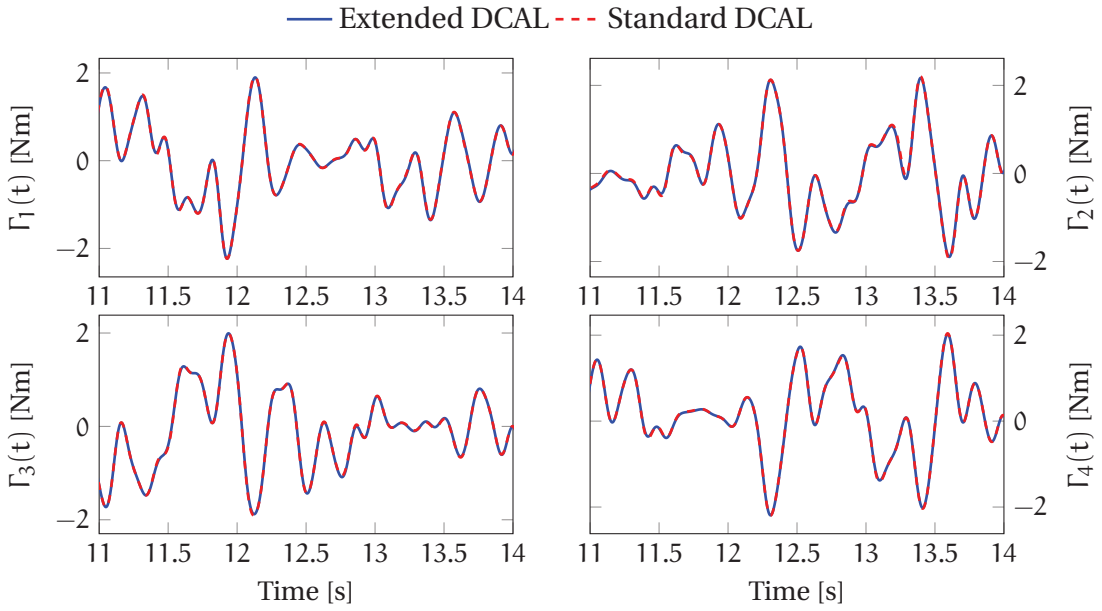


Figure 6.3 – Scenario 1 - Evolution of the control input torques Γ versus time.

start varying from the beginning of the trajectory unlike the mass parameter. This is only due to the absence of rotations in the starting of the trajectory.

The control inputs are depicted in Figure 6.3. For the sake of clarity, the plot is zoomed to the interval [11,14] seconds. We notice that the input torques generated by the proposed extended DCAL controller are slightly reduced compared to standard DCAL. This is another feature of the nonlinear gains that worth to be highlighted.

6.2.2 Scenario 2: robustness towards mass variation

In this scenario we added an extra payload to the traveling-plate of the robot with unknown mass and inertia. The motivation behind such a scenario is to test the robustness of the controller toward uncertainties. In a real application, the manipulator is expected to handle different payloads with different masses. Hence, it is up to the control scheme to adjust its parameters to keep the tracking errors as small as possible.

Figure 6.4 depicts the Cartesian tracking (zoomed at [11,14] seconds) errors of both controllers versus time. Similar to the first scenario, the tracking errors are significantly reduced thanks to the use of nonlinear feedback gains in the control law.

Table 6.3 summarizes the improvements brought by the proposed controller (up to 13.6%) with respect to the original one.

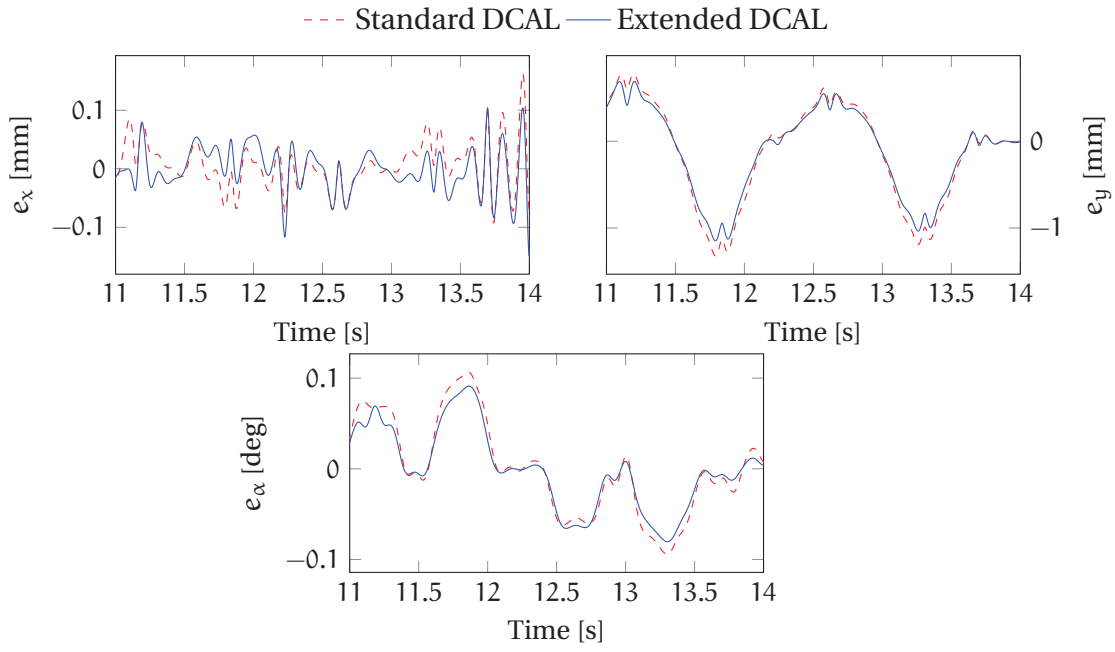


Figure 6.4 – Scenario 2 - Cartesian tracking errors

Table 6.3 – Scenario 2: Evaluation and comparison of the tracking performance.

	$RMSE_J$ [deg]	$RMSE_T$ [mm]	$RMSE_R$ [deg]
Standard DCAL	0.1276	0.48726	0.0412
Extended DCAL	0.1111	0.42056	0.0374
Improvement	12.93 %	13.6 %	9.2 %

The evolution of the estimated parameters for this scenario is shown in Figure 6.5. It can be observed that the adaptation algorithm adjusts the estimated parameters to newer steady state values. These new values correspond to the dynamic parameters of the traveling plate including the additional payload.

The generated control inputs are depicted in Figure 6.6 from which it can be observed slight reduction of the energy consumption for the extended DCAL compared to the standard one. Furthermore, it can also be seen that, for both controllers, the generated torques remain within the admissible region since they do not exceed their maximum allowed actuators' torques.

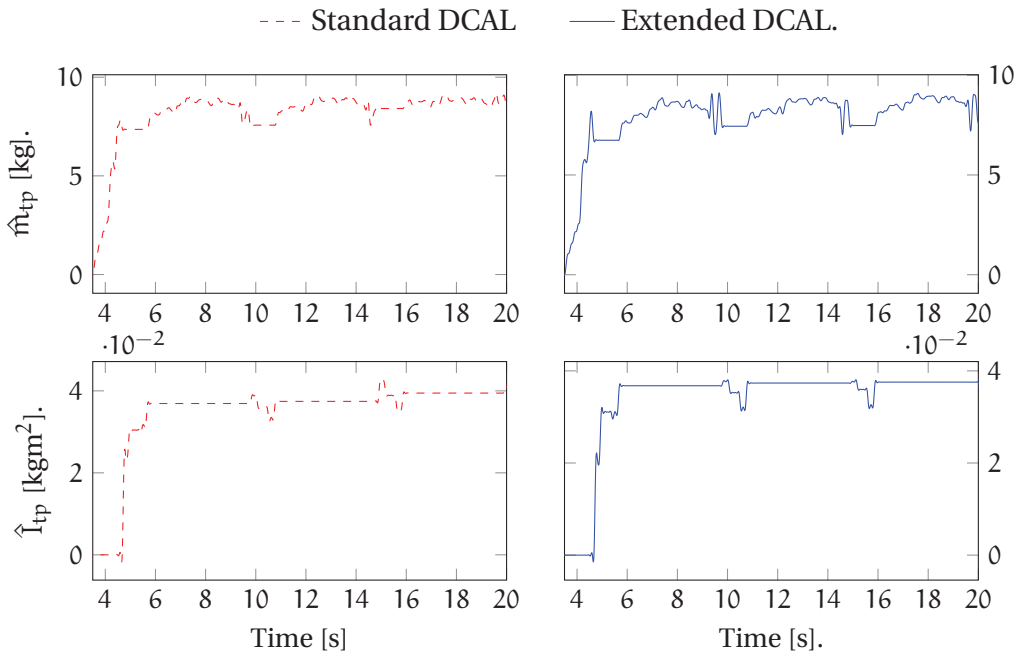


Figure 6.5 – Scenario 2 - Evolution of the estimated parameters

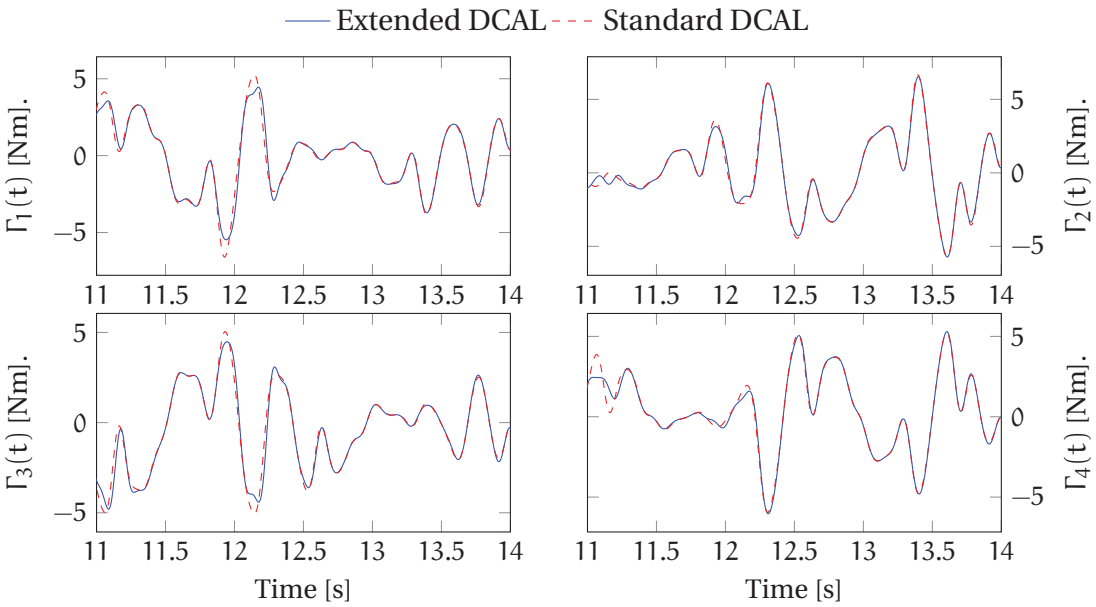


Figure 6.6 – Scenario 2 - Evolution of the control inputs

6.3 Experimental results of contribution 2 on Dual-V

To demonstrate the relevance of the proposed adaptive RISE controller and to show its superiority compared to standard RISE, real-time experiments were performed on the DUAL-V robot. Two scenarios were experienced in order to test the robustness of the controller, namely, a nominal scenario and a robustness scenario.

The controller parameters are shown in Table 6.4. Notice that we use the same parameters for both controllers. However, the adaptive RISE controller has an additional parameter K that corresponds to the estimation loop, namely, the adaptive gain. Those parameters

Table 6.4 – Parameters of the standard and adaptive RISE controllers

Standard RISE		Adaptive RISE	
Parameter	Value	Parameter	Value
α_1	40	α_1	40
k_s	25.5	k_s	25.5
α_2	124	α_2	124
β	75	β	75
		K	diag(75, 0.025)

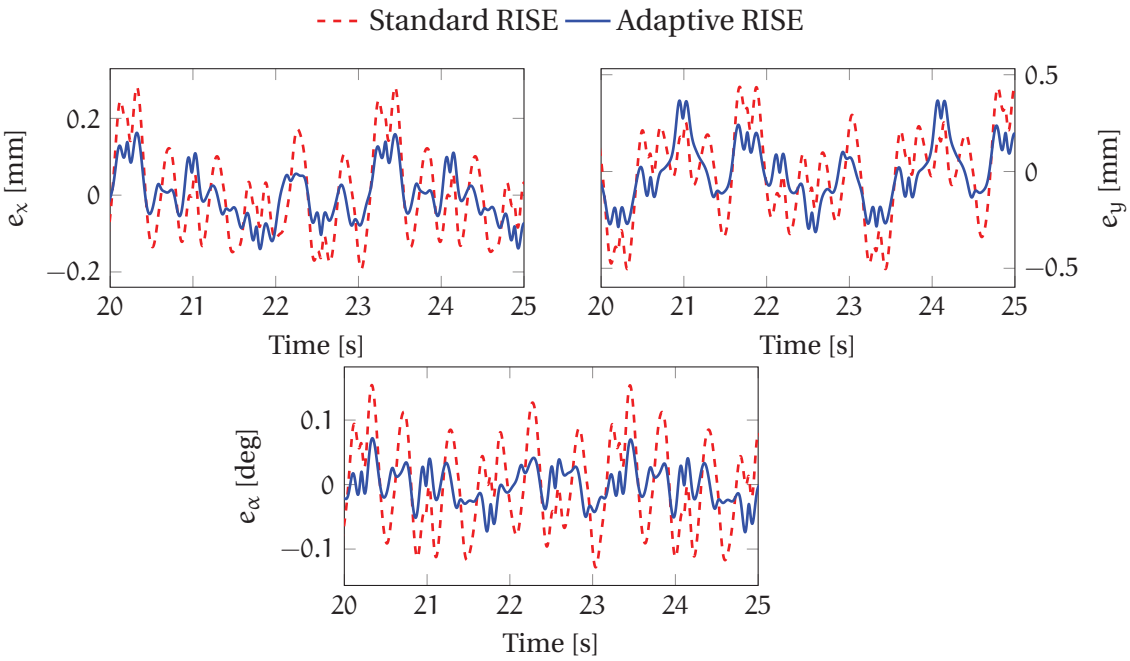


Figure 6.7 – Scenario 3 - Cartesian tracking errors

were obtained through trial and error technique since there is no systematic approach to tune the parameters of RISE control, yet. It is also worth mentioning that the same parameters are used for both scenarios; nominal one and robustness to mass variation one.

6.3.1 Scenario 3: nominal case

In this scenario, no additional mass is added to the mechanical structure. The Cartesian tracking errors are depicted in Fig 6.7 being zoomed on [11,14] seconds for clarity. One can clearly notice that the addition of the adaptive feedforward term in the proposed controller significantly reduces the tracking errors in all axes. Table 6.5 summarizes the performance of both controllers in terms of the RMS of the tracking errors.

The generated input control torques for both controllers of this scenario are depicted in

Table 6.5 – Scenario 3 - summary of the performance of both controllers.

	$RMSE_J$ [deg]	$RMSE_T$ [mm]	$RMSE_R$ [deg]
Standard RISE	0.2729	0.2515	0.0708
Adaptive RISE	0.2425	0.1652	0.0287
Improvement	11.14 %	34.31 %	59.46 %

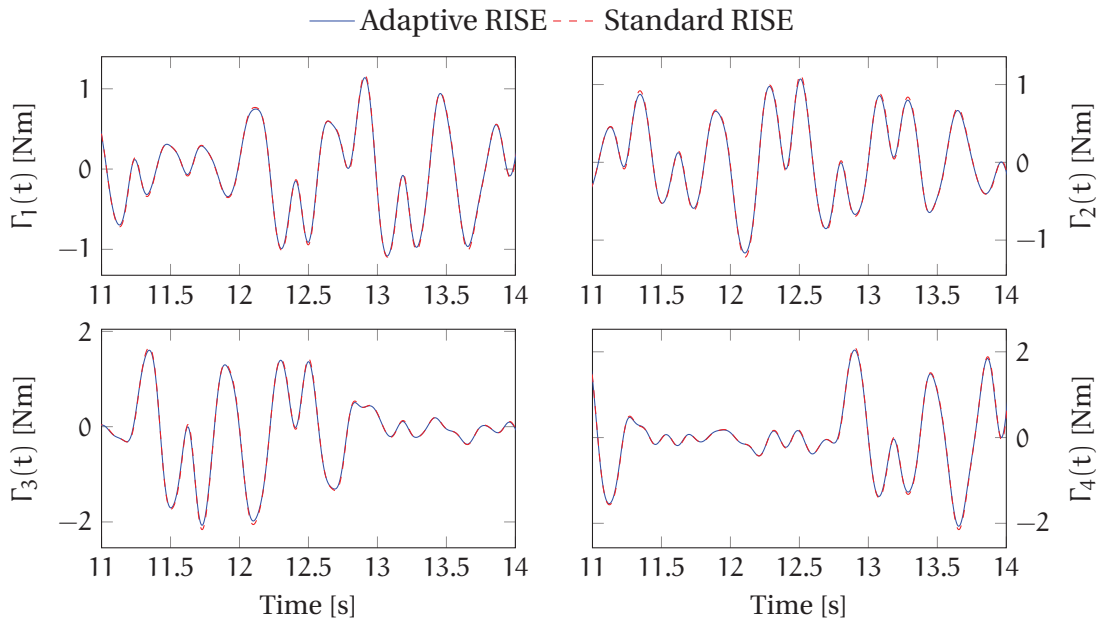


Figure 6.8 – Scenario 3 - Evolution of the control input torques versus time

Fig 6.8. We can notice that the proposed adaptive controller consumes slightly less energy than standard RISE.

The evolution of the unknown parameters estimates for the adaptive case are illustrated in Fig 6.9.

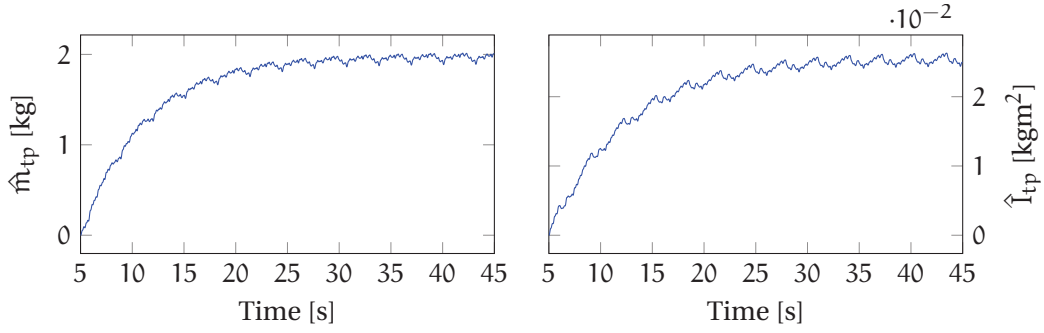


Figure 6.9 – Scenario 3 - Evolution of the estimated parameters of the adaptive RISE controller versus time

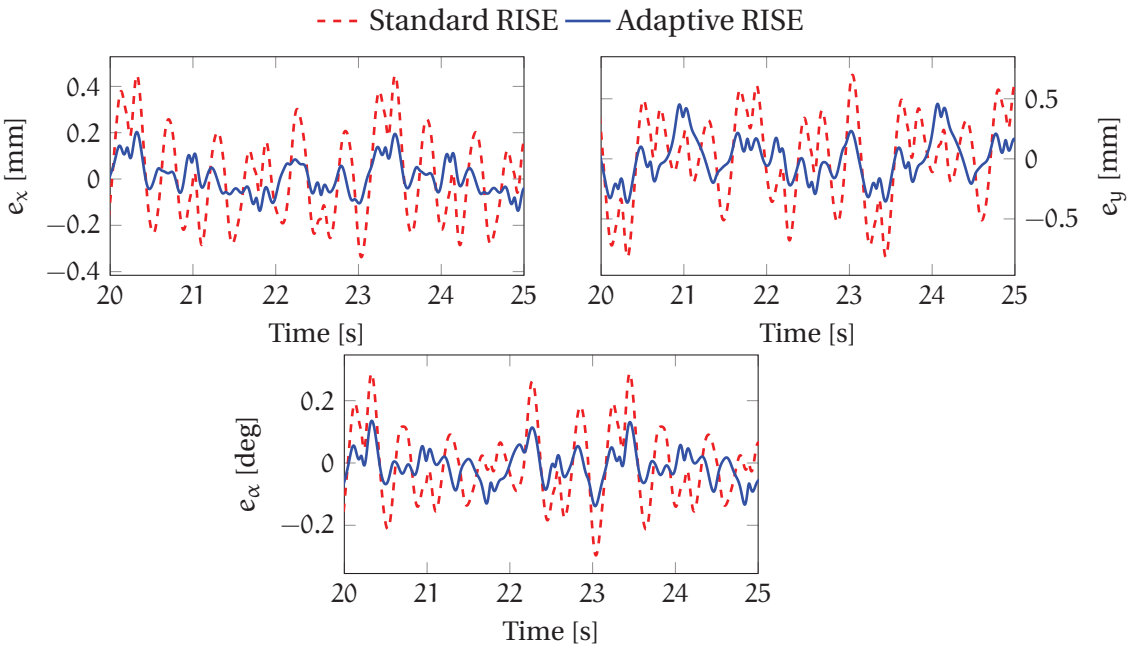


Figure 6.10 – Scenario 4 - Cartesian tracking errors.

6.3.2 Scenario 4: payload change

In this scenario, an additional unknown payload is added on top of the traveling-plate. The main motivation of experiencing this scenario is to first evaluate the robustness of the proposed adaptive controller to load changes. Then, to compare once again the RISE and adaptive RISE controllers in terms of tracking accuracy. The Cartesian tracking errors for this scenario being zoomed at [20,25] seconds are depicted in Fig 6.10. It can be clearly seen that the proposed adaptive controller yields much better tracking performance.

Table 6.6 summarizes the evaluated RMSE-based criteria for both controllers. It is obvious that a significant improvement is observed regarding the Cartesian tracking performance (up to 52 %).

The evolution of control input torques with respect to time are depicted in Fig 6.11. Fi-

Table 6.6 – Scenario 4 - summary of the performance of both controllers.

	$RMSE_J$ [deg]	$RMSE_T$ [mm]	$RMSE_R$ [deg]
Standard RISE	0.2492	0.4145	0.1247
Adaptive RISE	0.2396	0.1965	0.0592
Improvement	3.85 %	52.59 %	52.52 %

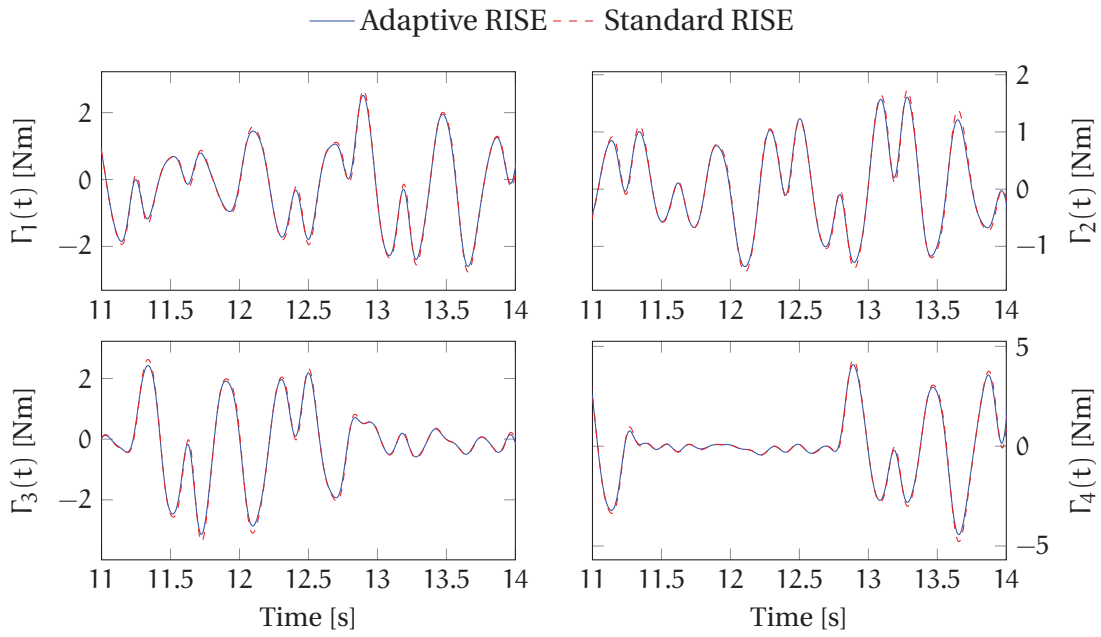


Figure 6.11 – Scenario 4 - Evolution of the control input torques versus time

nal conclusions can be therefore drawn that the proposed controller provides better tracking results while consuming slightly lower power.

Finally, the time evolution of the new estimated parameters due to the additional payload is illustrated in Fig 6.12. One can see that the parameters converge to new values different from those in the nominal case due to the unknown weight added on top of the traveling-plate.

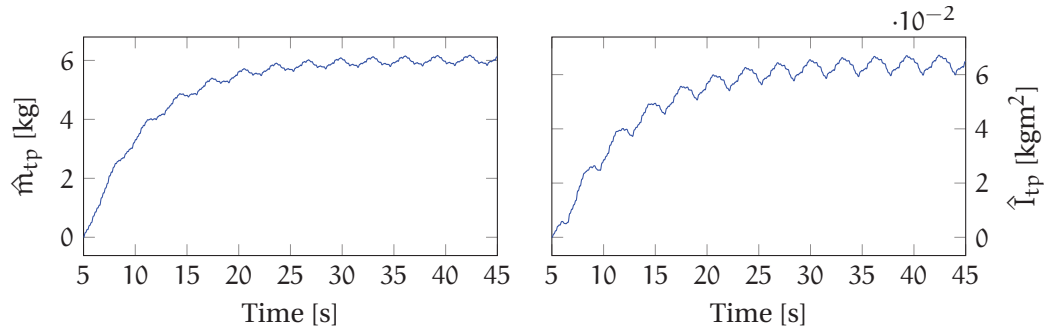


Figure 6.12 – Scenario 4 - Evolution of the estimated parameters of the adaptive RISE controller versus time

Table 6.7 – Parameters of the standard and adaptive RISE controllers

Standard RISE		Adaptive RISE	
Parameter	**Value**	**Parameter**	**Value**
α_1	600	α_1	600
k_s	0.43	k_s	0.43
α_2	0.66	α_2	0.66
β	2	β	2
		K	0.1

6.4 Experimental results of contribution 2 on Delta

To further demonstrate the relevance of the adaptive RISE controller, described in 3.3, on a different platform and a completely different control development system, we implement both standard and adaptive RISE controllers on the Delta 3-DOF robot. Similarly to the experiments performed on the DUAL-V robot, two scenarios were experienced; a nominal case and a robustness to mass variation case. The parameters of both controllers were obtained by trial and error, they are summarized in Table 6.7.

The desired trajectories used in this scenario are different from all the remaining ones which are generated according to the method described in 5.4. The desired trajectories used for this scenario are pick-and-place ones generated according to the method de-

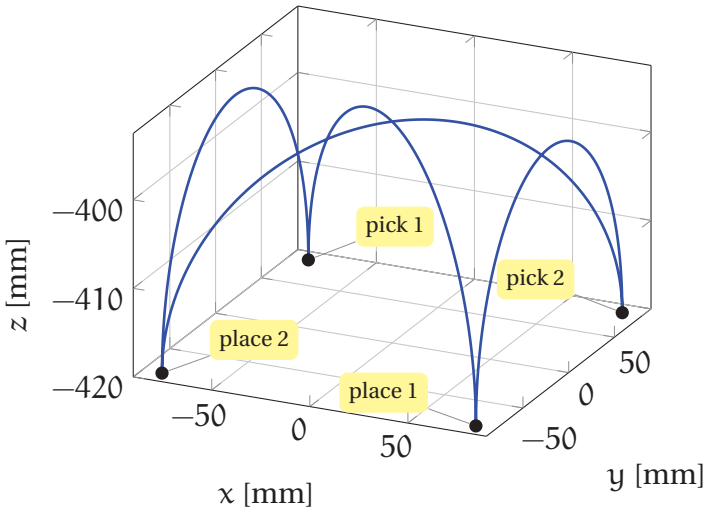


Figure 6.13 – Illustrative view of the experienced pick-and-place trajectory

Table 6.8 – Scenario 5 - summary of the performance of both controllers.

	$RMSE_J$ [deg]	$RMSE_T$ [mm]
Standard RISE	0.114	0.454
Adaptive RISE	0.068	0.262
Improvement	40.75 %	42.31 %

scribed in [Codourey, 1991]. This method consists of using semi-ellipses for the generation of the desired geometric motion. An illustrative 3D view of the pick-and-place trajectory used in this scenario is shown in Figure 6.13.

6.4.1 Scenario 5: nominal case

In this scenario, no additional payload is added to the traveling-plate of the Delta robot. The mass of the traveling-plate is supposed unknown and was initialized to zero ($\hat{m}_{tp}(0) = 0$). The adaptive term has then to estimate the real value of the mass of the traveling-plate in real-time to compensate for its dynamics and yield better tracking accuracy. Notice that the half-masses of the forearms of the robot are also included in $\hat{m}_{tp}$ (due to simplifying hypotheses of the dynamic model).

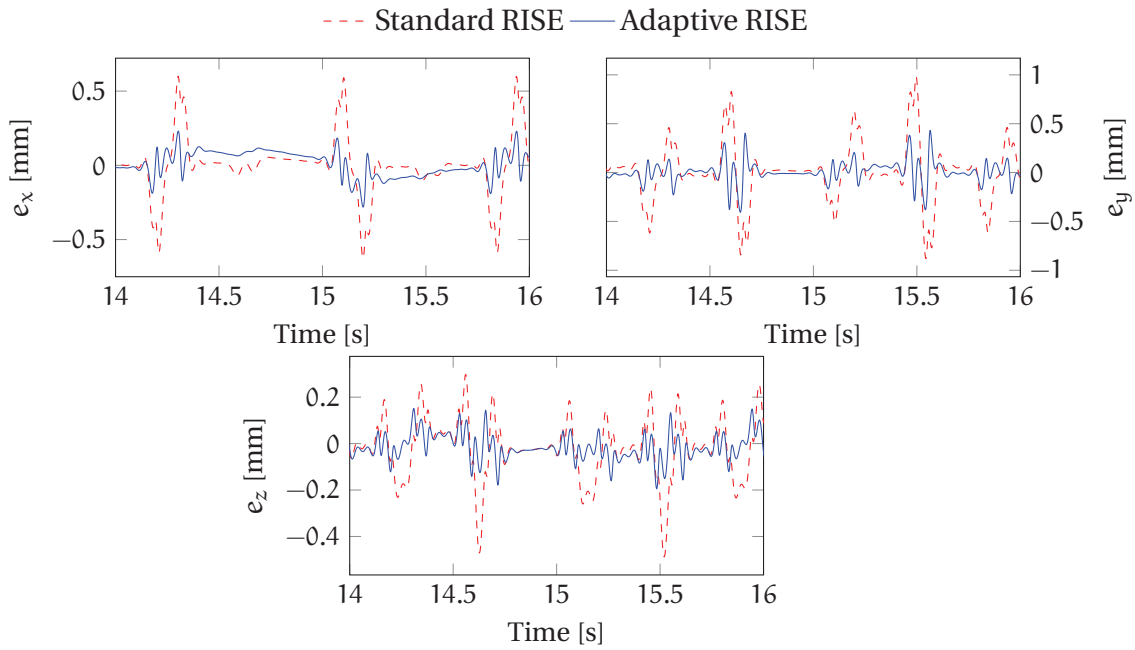


Figure 6.14 – Scenario 5: Cartesian tracking errors.

The Cartesian tracking errors in this scenario for both controllers are depicted in Figure 6.14. For the sake of clarity, the figures are zoomed in the interval $[14,16]$ seconds. It can be seen that the adaptive RISE controller performs much better than the non-adaptive one. We can also observe the presence of high amplitude spikes in the standard RISE controller that correspond to a stopping of the traveling plate at a the end of pick or place trajectory. These spikes are significantly reduced in the adaptive controller.

To quantify the enhancement of the tracking performance brought by the adaptive controller, the RMSE-based criteria introduced in 5.5 are evaluated for each controller. The obtained results are summarized in Table 6.8 where it can be seen observed that the improvement is very significant (up to 42 %).

The generated control input torques for both controllers are shown in Figure 6.15. For clarity, the plots are zoomed within the interval [14,16] seconds. It can be seen that both signals are in admissible ranges and do not show any high frequency components. It can also be observed that the control effort in the case of the adaptive controller are slightly reduced compared to the non-adaptive one.

Finally, the evolution of the estimated parameter $\hat{m}_{tp}$ versus time is shown in Figure 6.16. Keep in mind that this parameter corresponds to the mass of the traveling plate in addition to the three half-masses of the forearms.

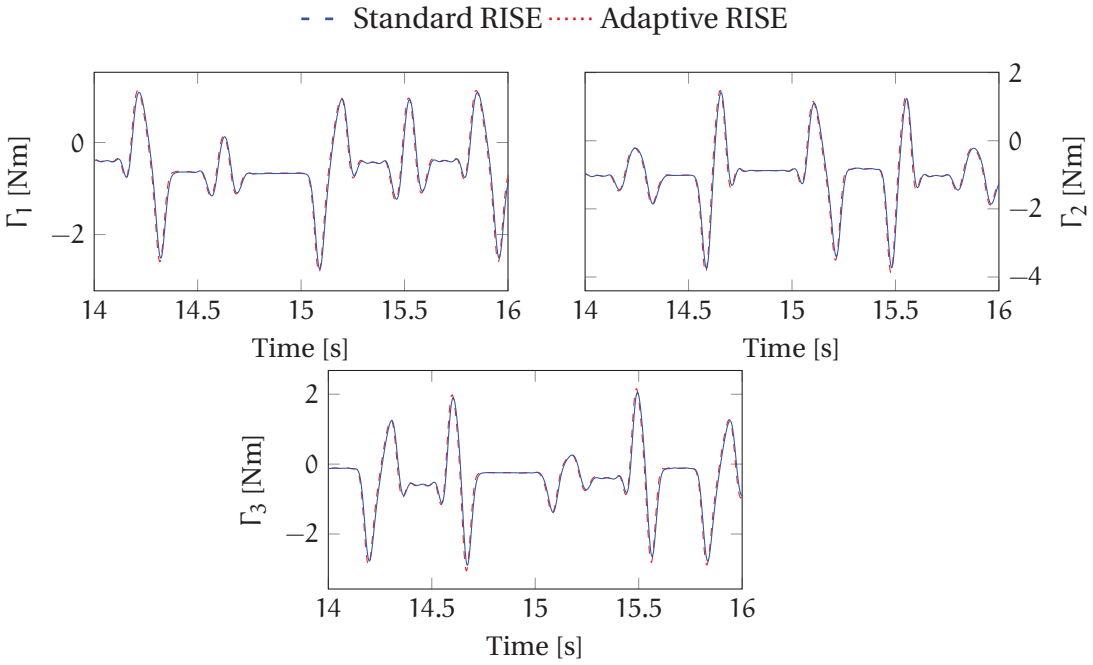


Figure 6.15 – Scenario 5: Evolution of the control inputs.

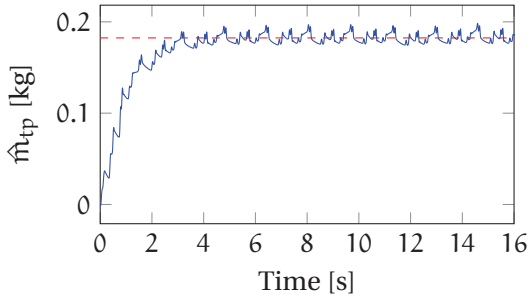


Figure 6.16 – Scenario 5: Evolution of the estimated traveling plate's mass versus time.

6.4.2 Scenario 6: robustness to mass variation

In this scenario, an additional payload of 120 g was added to the traveling-plate. Similarly to the previous scenario, the estimation loop has to keep updating the unknown parameter $\hat{m}_{tp}$ in real-time until it converges to its real value.

The Cartesian tracking errors for this scenario, which are the most relevant to show, are depicted in Figure 6.17. The plots are focused within the interval [14,16] seconds for clarity. It can be seen that, as expected, the tracking errors are significantly reduced in the case of

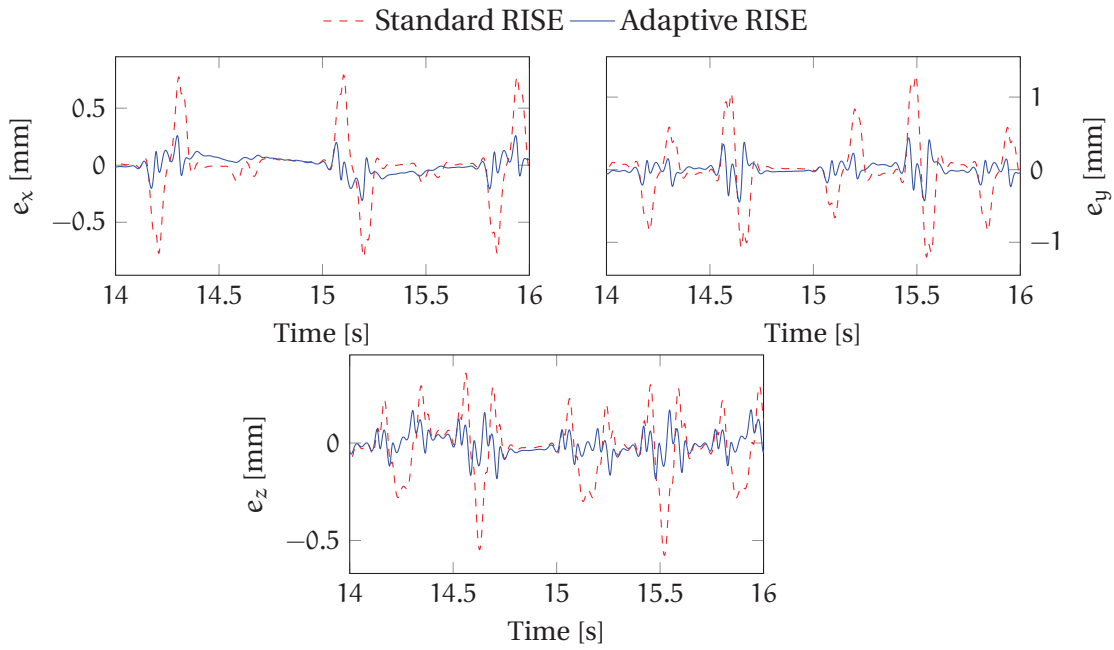


Figure 6.17 – Scenario 6: Cartesian tracking errors.

Table 6.9 – Scenario 6 - summary of the performance of both controllers.

	$RMSE_J$ [deg]	$RMSE_T$ [mm]
Standard RISE	0.147	0.593
Adaptive RISE	0.061	0.243
Improvement	58.79 %	59.12 %

the adaptive controller compared to the standard non-adaptive one. The improvement can be clearly seen at the occurrences of the spikes which correspond to the end of the current pick or place motion and the starting of the next one.

Table 6.9 summarizes the tracking performance in terms of the RMS of the errors. It can be noticed that the tracking errors are significantly reduced, even better than in the nominal scenario. Indeed, the tracking errors for this scenario are improved by more than 58 %.

The evolution of control input torques for both controllers is depicted in Figure 6.18. For clarity, the plots are zoomed within the time interval [14,16] seconds. Equivalently to the previous scenario, the control efforts are slightly reduced in the adaptive controller.

The evolution of the unknown adaptive parameter $\hat{m}_{tp}$, initialized to zero, is illustrated

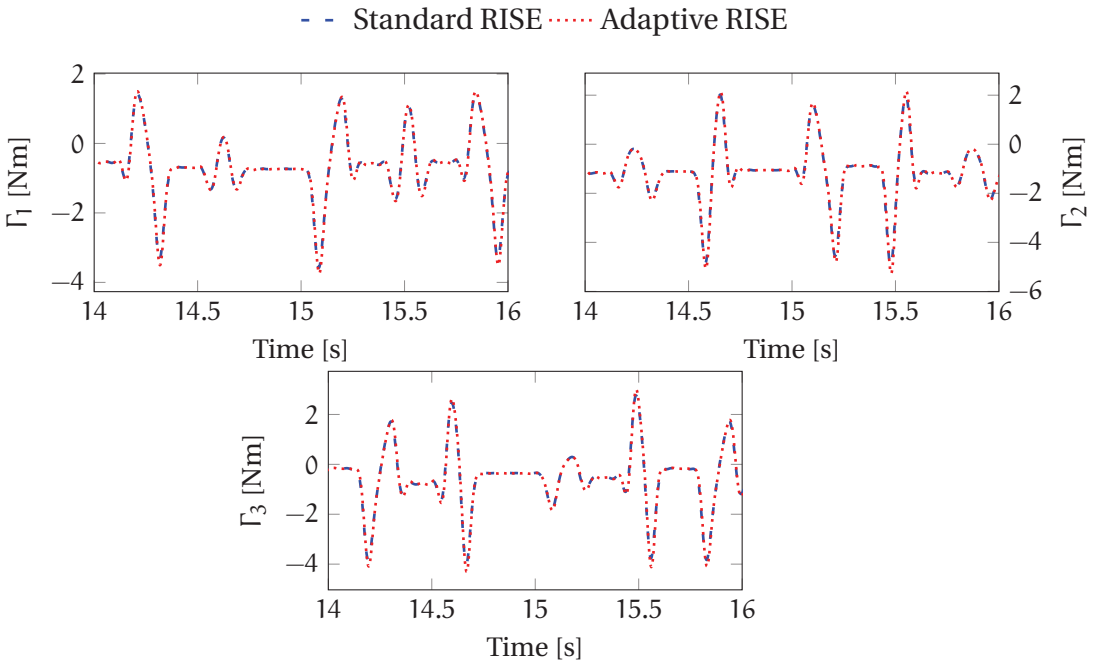


Figure 6.18 – Scenario 6: Evolution of the control inputs.

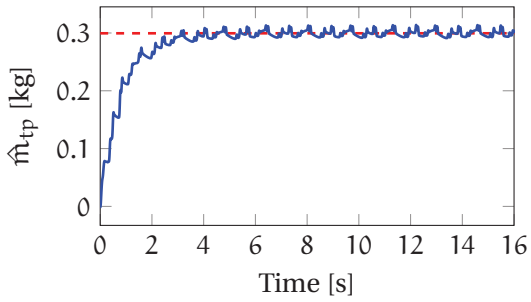


Figure 6.19 – Scenario 6: Evolution of the estimated traveling plate's mass versus time.

in Figure 6.19. Notice that the new estimated value is increased by 120 g compared to the previous scenario, which corresponds to the additional mass added to the traveling-plate.

6.5 Conclusion

In this chapter, real-time validation of the proposed controllers within **Solution 1** and experimental results were presented. Experiments have been conducted on two types of PKMs; the redundantly actuated DUAL-V robot and the non-redundantly actuated Delta robot.

First, the performance of the standard state of the art DCAL controller were compared to that of the proposed extended DCAL with nonlinear feedback gains. Experimental results obtained through real-time experiments on the DUAL-V robot have shown that the proposed controller outperformed the standard one in terms of tracking performance in both nominal and robustness scenarios. Furthermore, starting from unknown dynamic parameters of the robot, both controllers have managed to identify the real values of these parameters in real-time. The estimation capability of DCAL is an important feature that had to be preserved in the extended controller, despite the change of the feedback gains.

In the second part of this chapter, the performance of standard RISE and adaptive RISE controllers was compared through real-time experiments on the Delta and DUAL-V robots. The main motivation of this controller was to show that RISE could be augmented with the dynamics of the manipulator to achieve better tracking performance. Indeed, with both robots and through different case studies, the adaptive controller was superior than the standard one in terms of tracking of the reference trajectories.

In the next chapter, real-time experiments of the proposed control schemes within **Solution 2** are presented and discussed.

CHAPTER

7 Experimental validation of solution 2

Contents

7.1 Introduction

This chapter is dedicated for the real-time experimental results of the developed control schemes within **Solution 2** of this thesis. The first two controllers, namely, the $\mathcal{L}_1$ adaptive controller and the augmented $\mathcal{L}_1$ with nominal feedforward, will be validated on the VELOCE. The last developed controller within this thesis, namely, the $\mathcal{L}_1$ adaptive controller with adaptive feedforward dynamics, will be applied to the ARROW robot. Comparative study will be conducted based on the tracking performance of each controller to highlights its benefits.

7.2 Experimental results of contribution 3 on Veloce

In this section, the experimental results of the application of $\mathcal{L}_1$ adaptive control on the VELOCE robot are provided. For comparison purpose, the performance of the controller is compared with that of the PD controller. The choice of a PD controller for comparison is

Table 7.1 – Summary of the parameters of the $\mathcal{L}_1$ adaptive controller

$\mathcal{L}_1$ adaptive controller		PD controller	
Parameter	Value	Parameter	Value
Λ	$10 \times \text{diag}(65,65,65,65)$		
A_m	$\text{diag}(-6,-6,-6,-6)$		
θ_{max}	50	K_P	$10^2 \times \text{diag}(39,39,39,39)$
σ_{max}	30	K_D	$\text{diag}(6,6,6,6)$
Ξ	10^6		
Z	$10^3 \times \text{diag}(6,6,6,6)$		
$C(s)$	$144/(s^2+21.6s+144)$		

motivated by the fact that both controllers are non-model-based. In fact, it would be unfair to compare the $\mathcal{L}_1$ adaptive control with a controller that require the least of knowledge about the dynamics of the system since it is completely independent of any knowledge about the manipulator's dynamics.

The parameters of both, the PD and the $\mathcal{L}_1$ adaptive controllers, are summarized in Table 7.1. The PD parameters K_P and K_D as well as the parameters of of the feedback term

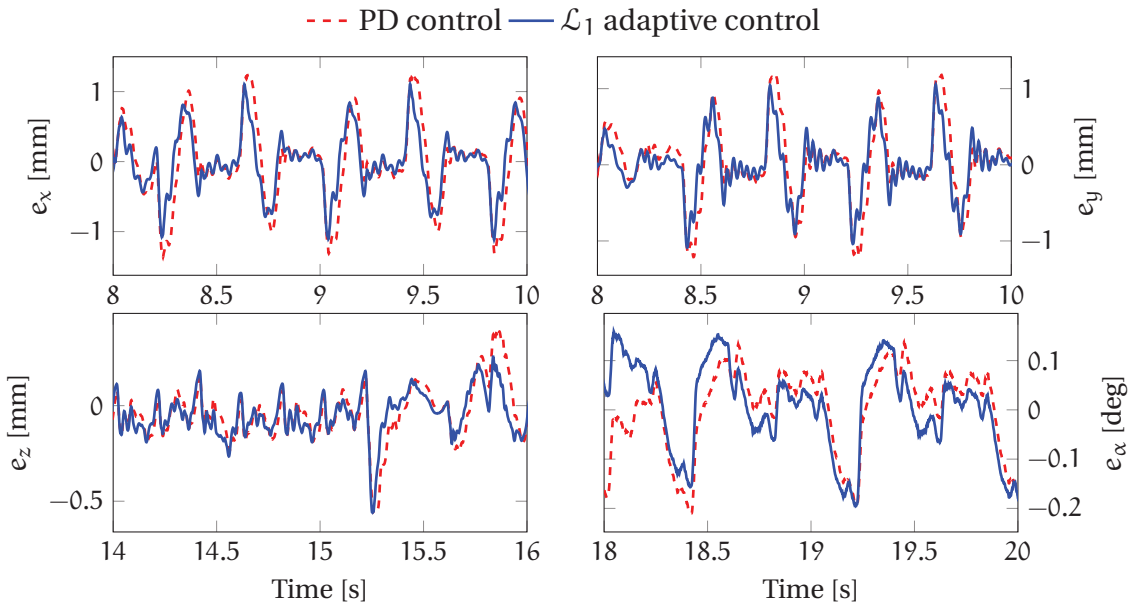


Figure 7.1 – Scenario 7: evolution of the Cartesian tracking errors versus time

Table 7.2 – Scenario 7: tracking performance comparison

	$RMSE_J$ [deg]	$RMSE_T$ [mm]	$RMSE_R$ [deg]
PD	0.1416	0.61988	0.0095
$\mathcal{L}_1$ adaptive control	0.1103	0.49885	0.0070
Improvement	22 %	20 %	26 %

of the $\mathcal{L}_1$ adaptive controller (Λ, A_m and Z) were obtained by trial and error. Regarding the projection bounds for the estimated functions (θ_{max} and σ_{max}), they were tuned in a way that they do not restrict the estimation too much while still respecting the actuators' limits. In fact, in our experiments, the estimated parameters never reach these values. Hence, they are just made for security measures. Finally, the adaptation gain Ξ is chosen large enough to increase the adaptation of the estimated parameters.

7.2.1 Scenario 7: nominal case

In this scenario, no additional payload or any disturbance are acting on the system. The robot in this case is in a typical experimental laboratory environment.

A comparison between both controllers in terms of Cartesian tracking performance is shown in Fig. 7.1. As it can be observed, the $\mathcal{L}_1$ adaptive controller outperforms the PD controller in each DOF. Notice that the plots are focused on a specific intervals that clearly highlight the superiority of the $\mathcal{L}_1$ adaptive controller.

The obtained results are compared in Table 7.2 in terms of the RMS of the tracking errors, from which the improvement brought by the proposed controller is quantified. The tracking errors are reduced by at least 20% with respect of those of the PD controller. On the other hand, the improvement is more important for the rotation of the traveling plate with an improvement of 26%.

For the $\mathcal{L}_1$ adaptive controller, the evolution of the estimated parameters $\hat{\theta}(t)$ and $\hat{\sigma}(t)$ is shown in Fig. 7.2 and Fig. 7.3 respectively. The estimated parameters keep oscillating and do not converge to a constant value since the desired trajectory of the traveling plate is time-varying. As expected, the estimated parameters never exceed their allowed bounds θ_{max} and σ_{max}. During the real-time experiments, the initial values $\hat{\theta}_0$ and $\hat{\sigma}_0$ were set to zero ($\hat{\theta}_0 = [0,0,0,0]^T$ and $\hat{\sigma}_0 = [0,0,0,0]^T$).

The generated control torques are shown in Fig. 7.4. It can be noticed that the inputs are away from the limits of the actuators, namely 127 Nm. Moreover, they are smooth and they do not show show any discontinuities.

7.3 Experimental results of contribution 4 on Veloce

To show the relevance of adding a feedforward term to the $\mathcal{L}_1$ adaptive controller, real-time experiments of the proposed control scheme in 4.4.3 were conducted on the VELOCE robot. Compared to the previous controller, the current one does not involve any additional parameters. Hence, the control parameters of the $\mathcal{L}_1$ adaptive controller summarized in Table 7.1 are used herein.

7.3.1 Scenario 8: nominal case

A comparison of the Cartesian tracking errors between the $\mathcal{L}_1$ adaptive controller and the proposed augmented one is depicted in Fig. 7.5. The plots are zoomed in the interval $[0,5]$ seconds for clarity. It can be clearly seen that the extended controller performs much better than the original $\mathcal{L}_1$ adaptive controller on all DOFs.

The obtained tracking results in terms of the RMSE are summarized in Table 7.3. It can be seen that the improvement is very significant regarding the joint tracking errors (up to 52%) and the translational movements (up to 68.8%) while only small improvement is observed on the rotation of the traveling plate (only 4.2%). This result highlights the benefits of using the dynamics of the manipulator in the control loop in terms of tracking performance.

The evolution of the estimated nonlinear functions $\hat{\theta}(t)$ and $\hat{\sigma}(t)$ versus time is depicted in Fig. 7.6 and Fig. 7.7 respectively. These figures clearly demonstrate the relevance

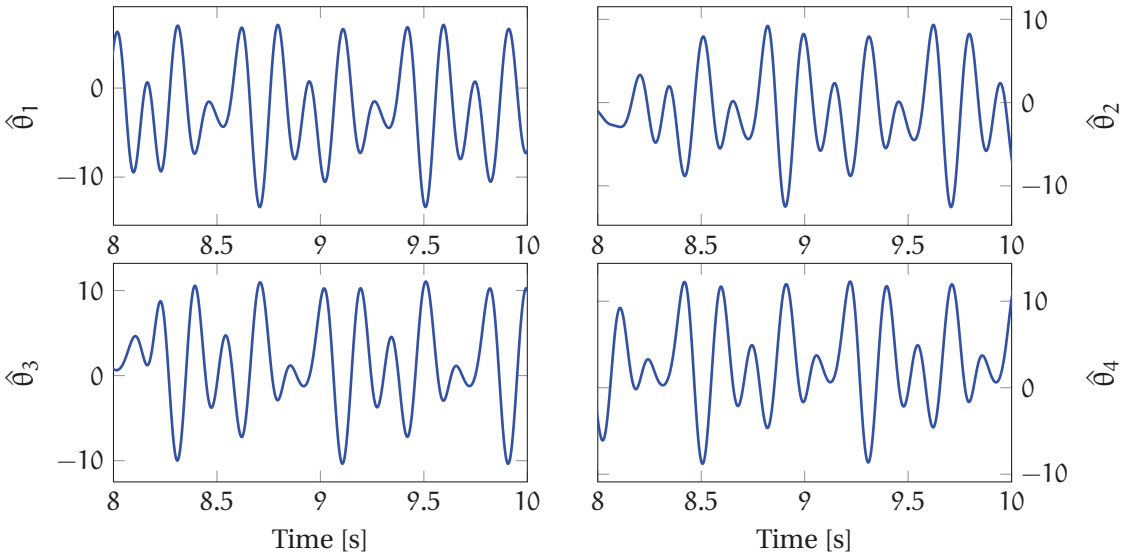


Figure 7.2 – Scenario 7: evolution of the estimated function $\hat{\theta}$ versus time.

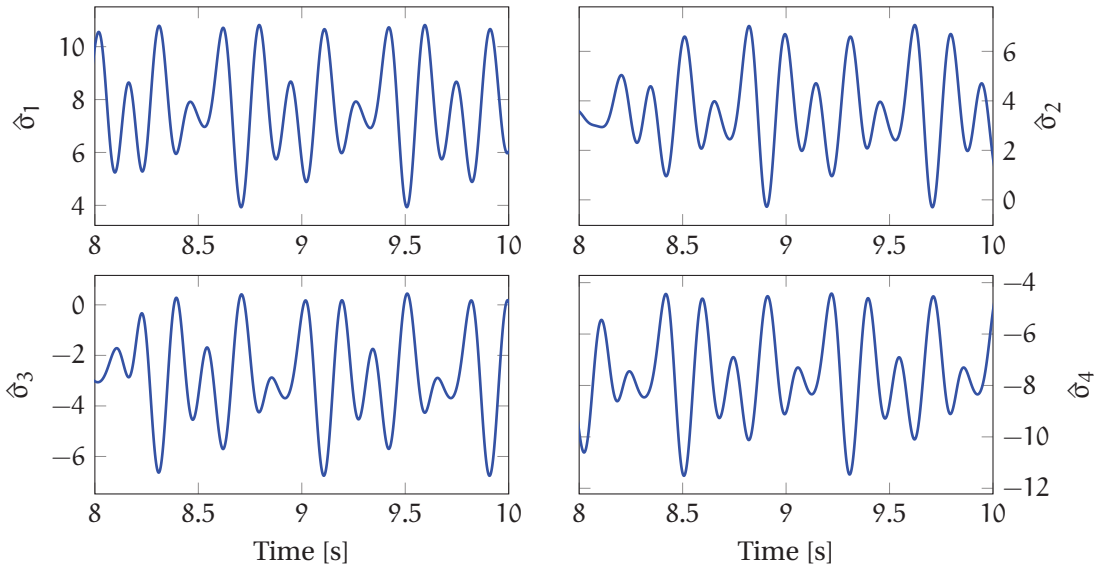


Figure 7.3 – Scenario 7: evolution of the estimated function $\hat{\sigma}$ versus time

of the proposed contribution. Indeed, the addition of the model-based feedforward considerably helps in compensating for the modeling inherent nonlinearities of the manipula-

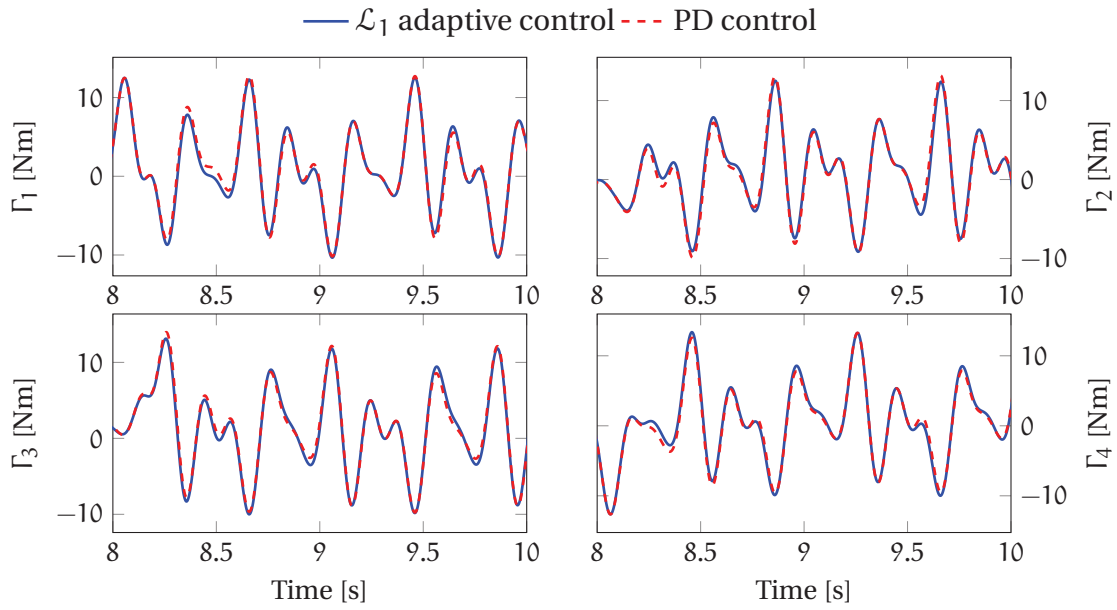


Figure 7.4 – Scenario 7: evolution of the adaptive $\mathcal{L}_1$ control inputs versus time

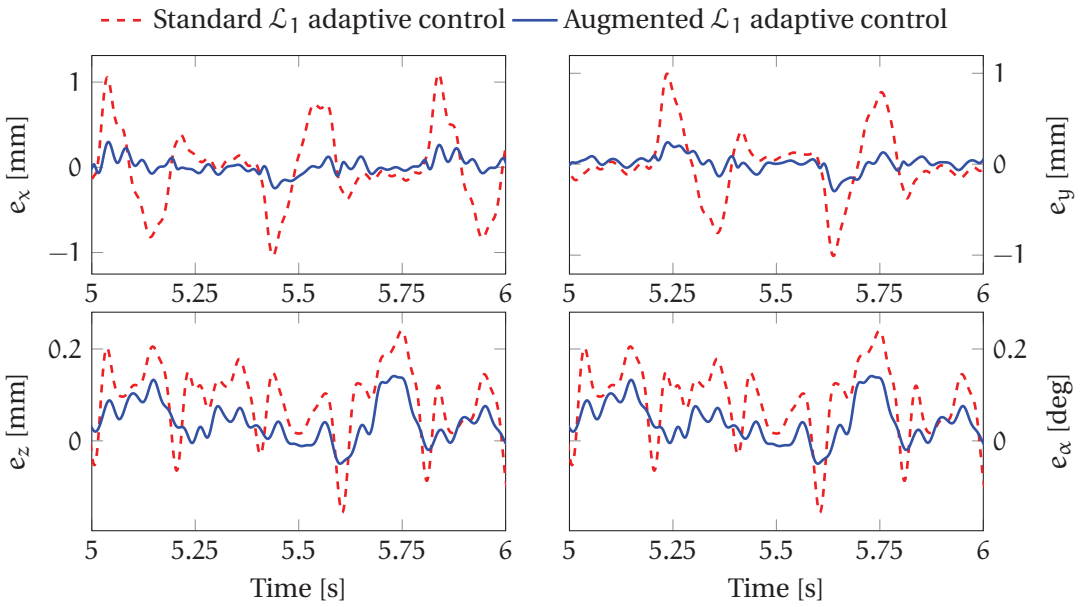


Figure 7.5 – Scenario 8: evolution of the Cartesian tracking errors versus time.

tor. Therefore, the remaining nonlinearities that have to be compensated for by the adaptive signal are of smaller amplitudes. This result is further illustrated in Fig. 7.9 where only the evolution of the adaptive components of the control law are plotted (i.e $\Gamma_{ad}(t)$). We can see that the adaptive signal, needed to compensate for the remaining nonlinearities, of the proposed controller is of lower amplitude than that of the conventional one.

The generated control inputs are shown in Fig. 7.8. The control inputs remain within the allowable range and do not exceed the limits of the actuators. The amplitudes of the signals in both controllers are roughly similar. Hence, the following result can be drawn: the proposed extension improves the tracking performance of the closed-loop system while consuming almost the same energy as the conventional controller.

Table 7.3 – Scenario 8: tracking performance comparison

	RMS_J [deg]	RMS_T [mm]	RMS_R [deg]
$\mathcal{L}_1$-AC	0.1012	0.4429	0.0082
Augmented $\mathcal{L}_1$-AC	0.0486	0.1391	0.0078
Improvement	52 %	68.6 %	4.2 %

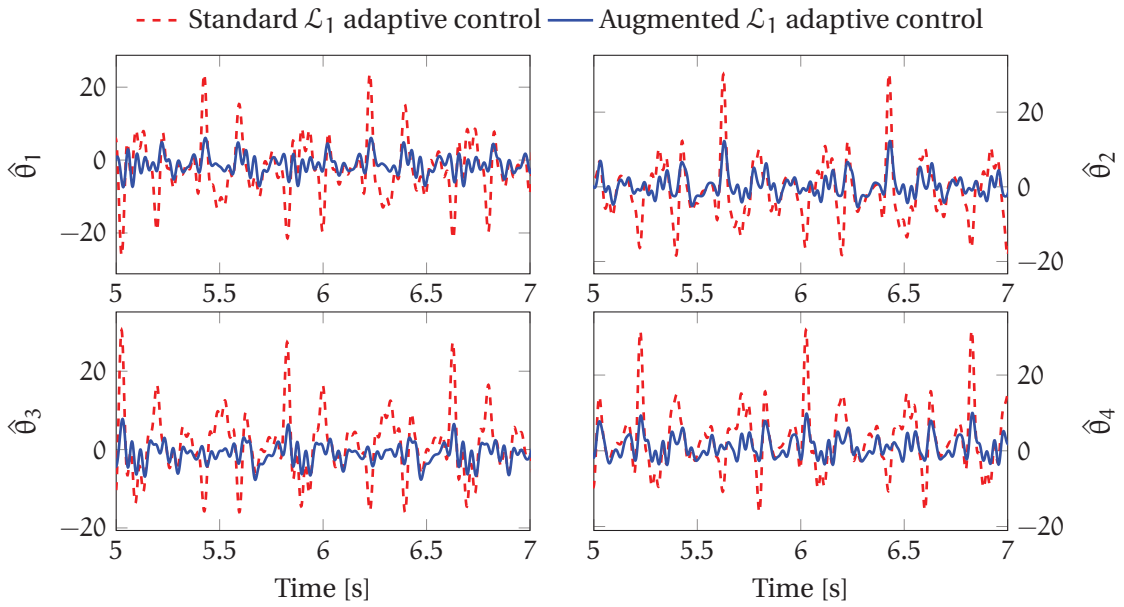


Figure 7.6 – Scenario 8: evolution of the estimated parameter $\hat{\theta}$ versus time.

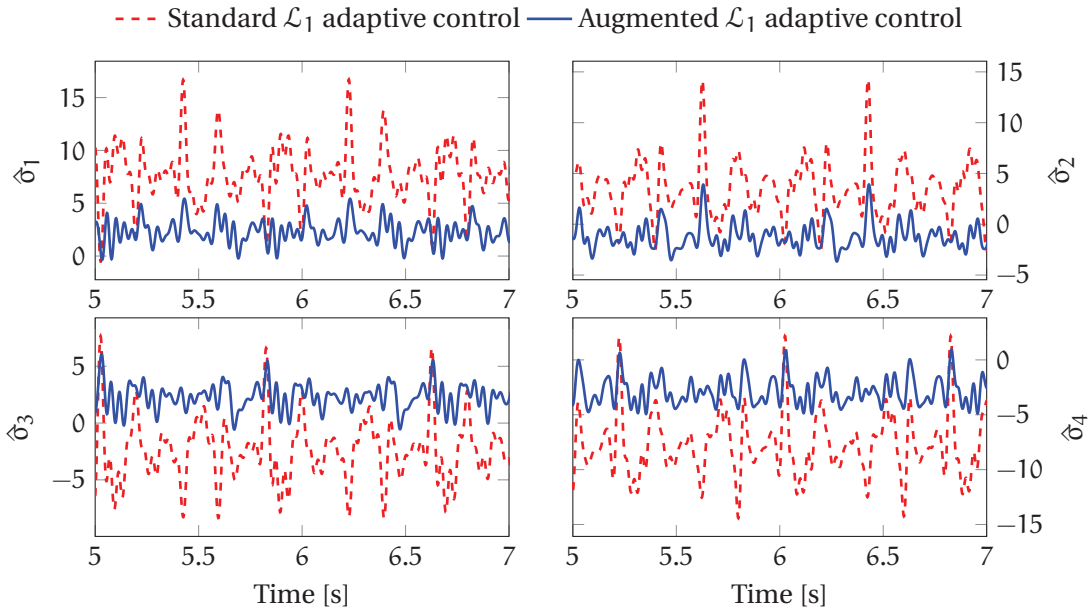


Figure 7.7 – Scenario 8: evolution of the estimated parameter $\hat{\sigma}$ versus time.

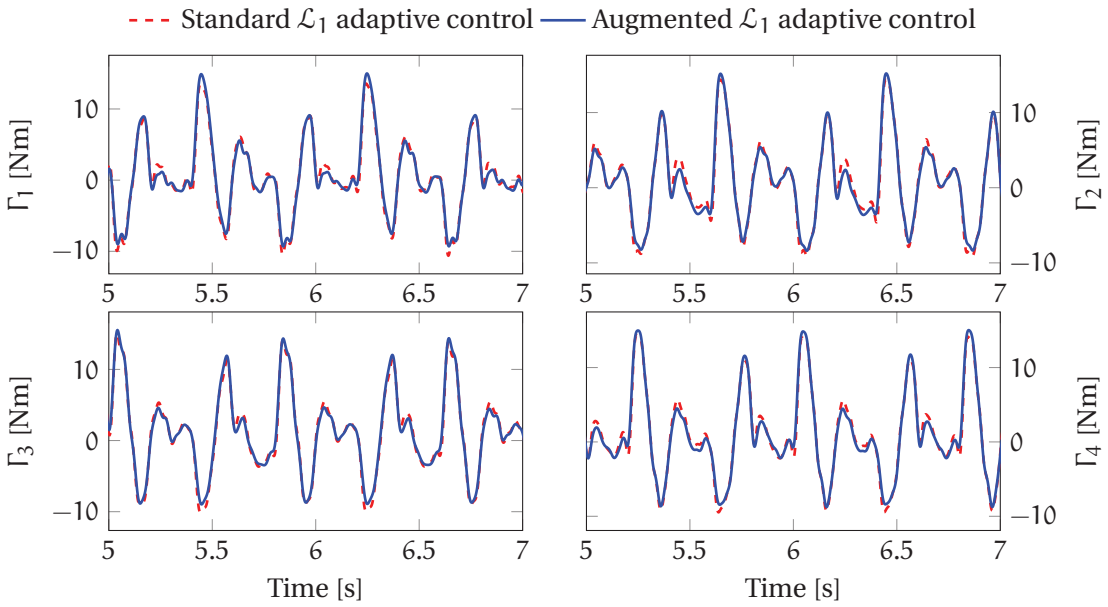


Figure 7.8 – Scenario 8: evolution of the control input torques Γ versus time.

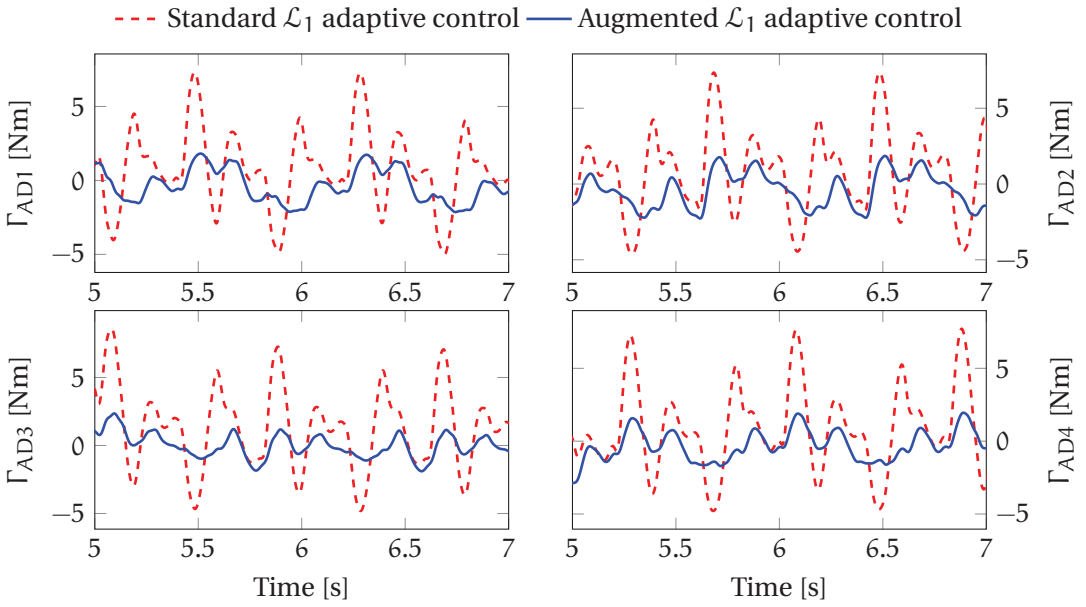


Figure 7.9 – Scenario 8: evolution of the adaptive control input torques Γ_{AD} versus time.

Table 7.4 – Summary of the control parameters used in real-time experiments

Standard $\mathcal{L}_1$ adaptive control		$\mathcal{L}_1$-ac with adaptive feedforward	
Parameter	Value	Parameter	Value
λ	725	λ	725
A_m	$-235 \times I_{6\times6}$	A_m	$-235 \times I_{6\times6}$
θ_b	35	θ_b	35
σ_b	35	σ_b	35
Ξ	10^6	Ξ	10^6
Z	$5765 \times I_{6\times6}$	Z	$5765 \times I_{6\times6}$
$C(s)$	$144/(s^2+16.8s+144)$	$C(s)$	$144/(s^2+16.8s+144)$
		K	diag([1,1,1,0.005])

7.4 Experimental results of contribution 5 on Arrow

To demonstrate its relevance, the proposed controller in 4.5.3 was implemented on the ARROW robot. This controller, which is based on $\mathcal{L}_1$ adaptive control, deals with both modeled dynamics and non-modeled ones adaptively. Hence, it is supposed that it provides better tracking performance compared to standard $\mathcal{L}_1$ adaptive control. The controller parameters were obtained in a similar manner to those in 6.2.2. They are summarized in Table 7.4. Notice that the proposed controller has one additional control parameter K that corresponds to the model parameters adaptation diagonal matrix.

7.4.1 Scenario 9: nominal case

A comparison between the $\mathcal{L}_1$ adaptive controller and the proposed one in terms of Cartesian tracking errors is illustrated in Figure 7.10. It can be seen that the proposed model-based augmented controller clearly provides much better tracking than the original controller. Thanks to the model-based adaptive term, the controller successfully compensates the modeled nonlinear dynamics with initially unknown parameters.

The obtained results are summarized in Table 7.5, where the improvements brought by

Table 7.5 – Scenario 9: tracking performance comparison in terms of RMS errors

	RMS_J [mm]	RMS_T [mm]	RMS_R [deg]
$\mathcal{L}_1$ adaptive control	0.0496	0.0343	4.7×10^{-3}
Extended $\mathcal{L}_1$ adaptive control	0.0102	0.0074	1.2×10^{-3}
Improvement	79.43 %	78.59 %	74.1

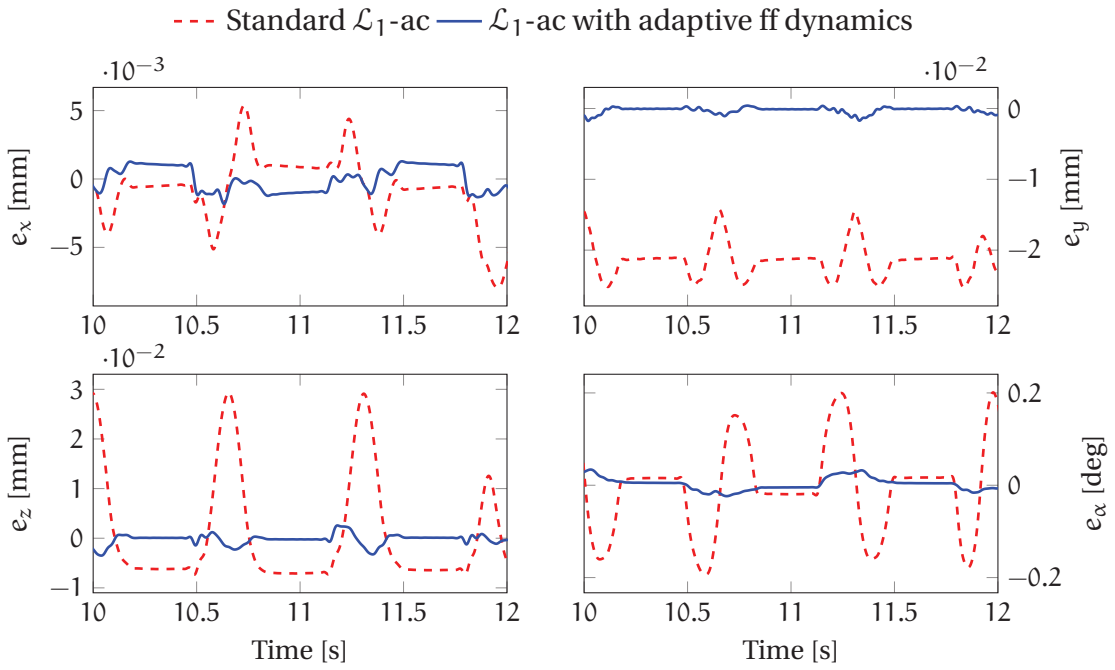


Figure 7.10 – Scenario 9: evolution of the Cartesian tracking errors.

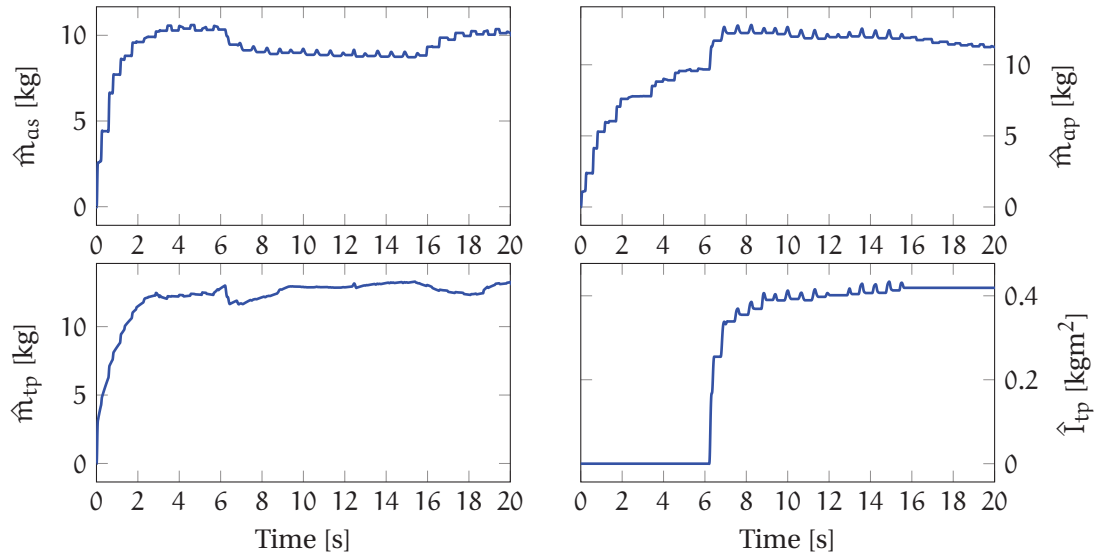


Figure 7.11 – Scenario 9: evolution of the estimated parameters of the adaptive feedforward term.

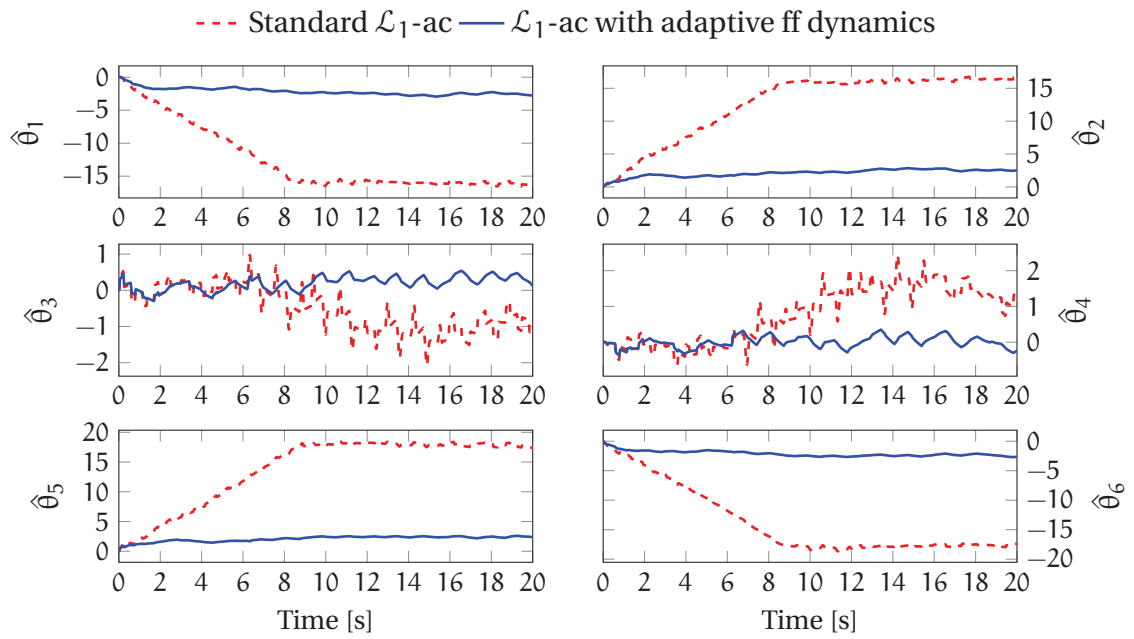


Figure 7.12 – Scenario 9: evolution of the estimated function $\hat{\theta}(t)$.

the proposed controller are clearly highlighted. Indeed, the joint tracking errors were reduced by 79.4 % and the Cartesian ones by up to 78.6 %. The improvements of the tracking errors are achieved thanks to the better compensation of the nonlinearities of the model. This means that the estimated model parameters do converge closer to their real values, otherwise the tracking performance would have been worst. Indeed, as it can be seen in Figure 7.11, the estimated parameters, initialized to zero, converge to their real values reported in Table 5.4. It is noticed that the rotational inertia of the platform is not adjusted immediately which is due to the pure translational motion at the beginning (i.e. $t \in [0, 7]$ s).

Furthermore, since the model-based nonlinearities are appropriately compensated, the estimated parameters forming the adaptive term $\Gamma_{ad}(t)$ would be smaller in the case of the proposed controller. Indeed, this interesting results can clearly be observed in Figures 7.12 and 7.13, where we can see smaller amplitudes for the estimated parameters in the case of the proposed controller compared to the original one. This means that the structured uncertainties were better compensated by the proposed additional term and hence, the $\mathcal{L}_1$ adaptive term should account for reduced nonlinearities.

Finally, the control input forces generated by both controllers are shown in Figure 7.14.

7.5 Conclusion

In this chapter, real-time experiments of the proposed controllers in the context of **Solution 2** have been performed. Similar to the previous chapter, we used two different platforms for validation. The first one is the VELOCE non-redundantly actuated PKM, while the second one is the ARROW redundantly actuated robot.

First, $\mathcal{L}_1$ adaptive control has been validated for the first time on a parallel manipulator, namely, the VELOCE robot. Then, to demonstrate the relevance of this control strategy regarding its tracking capabilities, a PD controller was implemented simultaneously to compare its performance with that of $\mathcal{L}_1$ adaptive control. As expected, the adaptation capabilities of the $\mathcal{L}_1$, which require no knowledge of the dynamics of the system, allowed to achieve better tracking performance than the PD controller.

Later on, the relevance of adding a nominal dynamic-based loop to the non-model-based $\mathcal{L}_1$ adaptive control was verified through real-time experiments on the VELOCE robot. Compared to standard $\mathcal{L}_1$ adaptive control, the proposed augmented controller has shown a net superiority in terms of tracking performance, both in joint and Cartesian spaces. This result was the first step towards developing a new class of controllers taking into consideration both the structured and unstructured uncertainties adaptively.

Finally, the developed $\mathcal{L}_1$ adaptive controller with adaptive dynamics has experimentally applied on the ARROW robot. To show its relevance, its performance was compared

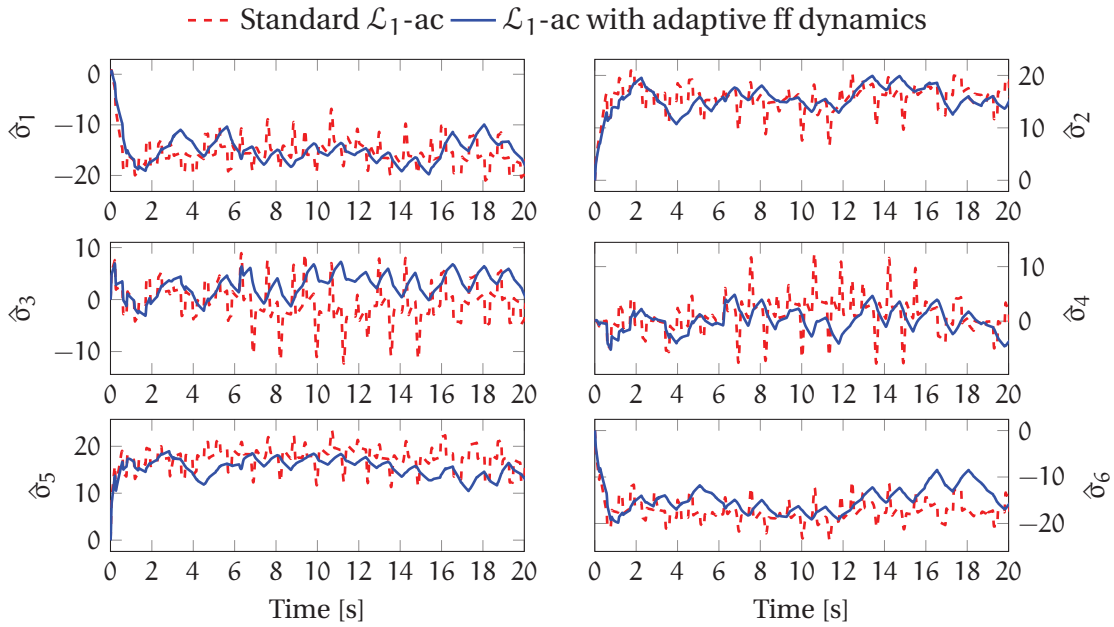


Figure 7.13 – Scenario 9: evolution of the estimated function $\hat{\sigma}$ versus time.

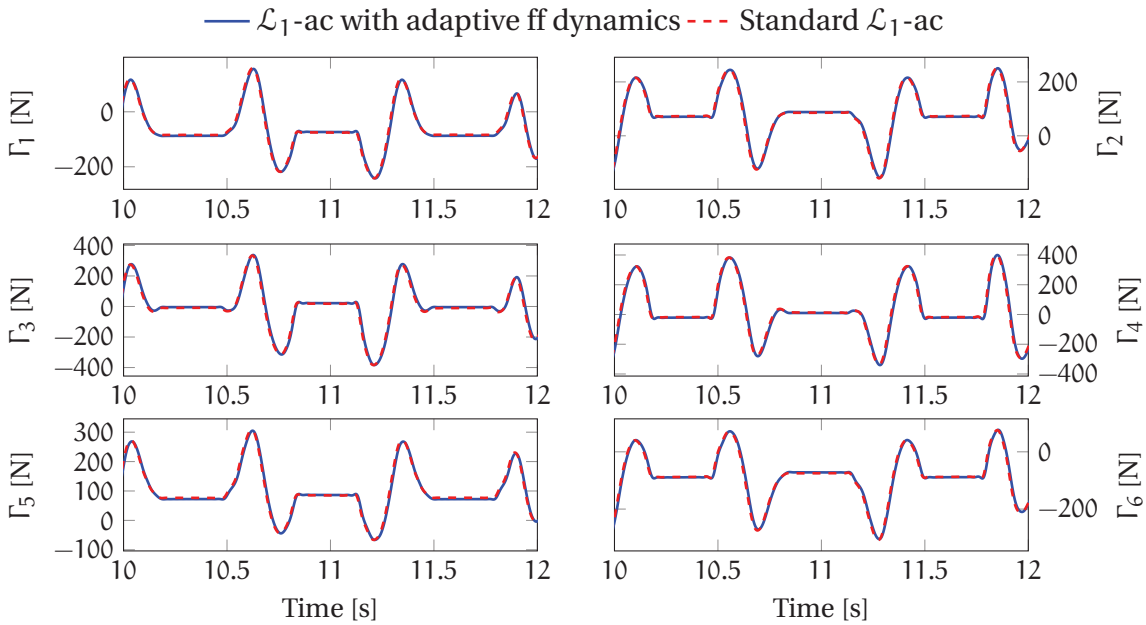


Figure 7.14 – Scenario 9: evolution of the input control forces versus time.

to that of the original $\mathcal{L}_1$ adaptive controller. It was shown that the proposed controller performs much better than the standard one. Moreover, the proposed adaptation strategy allowed to estimate the dynamic parameters of the AROW robot, initially supposed unknown.

General conclusion

The objective of this thesis was to design advanced control schemes for parallel manipulators to enhance their tracking capabilities. The developed strategies fall in the context of a national research project called ARROW. This project aims at improving the dynamic performance of parallel manipulators. More precisely, the objective of the project from control point of view is to design control schemes that enable accurate tracking of fast reference trajectories.

Summary of the work

After a thorough study and analysis of existing state of the art control schemes, adaptive controllers were found to be the most adequate solution to achieve the specified control objective. Such strategies are known by their estimation capabilities of the various uncertainties that may deteriorate the control performance and hurt its tracking capabilities. In this context, various joint-space control solutions were proposed throughout this thesis. The choice of designing control schemes in joint space instead of Operational space is motivated by the absence of external sensors in most industrial manipulators. Furthermore, the easiness of the inverse kinematic problem in the case of parallel manipulators reinforces the adopted choice. The proposed controllers that have been proposed in this work are as follows

1. An extended adaptive DCAL controller with nonlinear time-varying feedback gains with enhanced tracking performance. This controller was motivated by the various state of the art applications of nonlinear gains in conjunction with non-adaptive model-based controllers. As a matter of fact, this was the first time ever that nonlinear feedback gains are used in an adaptive control scheme. As a consequence,

the proposed extended DCAL strategy inherits the benefits of both DCAL control and nonlinear feedback gains. Moreover, to deal with internal efforts which may be dangerous for the mechanical structure, a projection method is implemented to reduce their effects. Real-time experiments on a 4-DOF redundantly actuated parallel manipulators demonstrated our claims. The proposed controller outperformed the standard DCAL with fixed feedback gains in terms of tracking performance, in nominal and robustness scenarios.

2. Extending a robust control strategy called RISE and endowing it with adaptation capabilities in order to efficiently compensate for modeling uncertainties. The standard RISE control law features a *signum* function, typically used in robust control techniques, that accommodates for a large class of uncertainties. Inspired from some extensions of RISE in the literature, we have proposed to include a model-based adaptive feedforward. This choice is motivated by the desire of improving the tracking performance of the controller. Indeed, the additional term efficiently compensates for the modeling structured uncertainties. The remaining ones are accounted for by means of the RISE control term. Real-time experiments on the DUAL-V robot have highlighted the relevance of the proposed solution. As it was expected, the adaptive controller provided better tracking performance than the standard one. Moreover, to show that the proposed controller can be applied on both redundant and non-redundant manipulators, experiments were conducted on the Delta robot at EPFL in Switzerland. The provided results were compliant with what was obtained in the case of the DUAL-V robot.
3. Application of the recent $\mathcal{L}_1$ adaptive control for the first time on a parallel manipulator. This strategy is best known for its decoupled estimation and control loops enabling fast adaptation and guaranteed robustness. The performance of $\mathcal{L}_1$ adaptive control was evaluated through real-time experiments on a 4-DOF parallel manipulator called VELOCE. Unfortunately, only a small improvement of the tracking performance was observed compared to PD control.
4. To improve the performance of $\mathcal{L}_1$ adaptive control, we have proposed to augment it with a nominal feedforward term based on the inverse dynamics of the system. This choice was inspired from conventional model-based controllers where a dynamic loop is usually appended to the feedback term. Real-time experiments conducted on VELOCE robot highlighted the outcomes of the proposed augmented controller with respect to its standard one. Especially, the tracking errors were significantly reduced thanks to the added model-based control term which compensates for the model nonlinearities.
5. An extended $\mathcal{L}_1$ adaptive controller with adaptive dynamics that considers both modeled and non-modeled uncertainties. Since the benefits of including the dynamics of the manipulator in the control loop were clearly demonstrated in the previous controller, we further extend this idea to adjust the dynamics loop in real-time.

Such solution allows to extend the capabilities of the controller to deal with scenarios where the parameters of the manipulator are unknown, uncertain or time-varying. Experimental results have clearly shown the outcomes of the proposed strategy. Particularly, compared to standard $\mathcal{L}_1$ adaptive control, the tracking errors were significantly improved when applied on the ARROW robot. Furthermore, the dynamic parameters of the robot, initially supposed unknown, were successfully adjusted to their real values.

Perspectives

In this thesis, multiple control schemes for parallel manipulators have been proposed targeting one main problem: enhancing the tracking performance of fast reference trajectories. We have shown that existing controllers can be revisited with intuitive techniques in the aim of reducing the tracking errors.

Possible directions to continue and extend this thesis may include

- Applying all the proposed control schemes within this on the ARROW robot and make a fair comparison between them considering various scenarios that could be encountered in practice (uncertainties disturbances).
- Investigating the use of other nonlinear gains in model-based adaptive controllers. In fact there exist a variety of nonlinear feedback gains proposed in the literature. Hence, it would be interesting to run them into tests to figure out which one yields the best performance.

APPENDIX

A Stability analysis of the extended DCAL controller

Theorem A.1 *The joint position and velocity tracking errors ($e(t)$ and $\dot{e}(t)$, respectively) of a mechanical manipulator whose dynamics is governed by (3.1), with $\tau_d = 0$, controlled by (3.11) goes to zero as time goes to infinity, provided that the control design parameters α_p and α_v are chosen such that $\alpha_p > 1$ and $\alpha_v > 1$ and that their lower bounds K_{P_m} and K_{V_m} in (3.14) and (3.15), respectively, are chosen large enough.*

Proof. To analyze the stability of the PKM in closed-loop with the proposed extended DCAL controller, let us first rewrite its dynamics (3.1) in terms of $e(t), \dot{e}(t)$ and $e_v(t)$ as follows

$$M(q)\dot{e}_v = M(q)(\ddot{q}_d + \lambda\dot{e}) + C(q,\dot{q})(\dot{q}_d + \lambda e) - C(q,\dot{q})e_v + G(q) - \Gamma \tag{A.1}$$

By adding and subtracting $Y(q_d, \dot{q}_d, \ddot{q}_d)\Phi$ to (A.1), we obtain

$$M(q)\dot{e}_v = -C(q,\dot{q})e_v + \tilde{Y}(.)\Phi + Y(q_d, \dot{q}_d, \ddot{q}_d)\Phi - \Gamma \tag{A.2}$$

where $\tilde{Y}(.)$ is the error in the dynamics emerging from using the desired trajectories instead of the measured ones. It given by:

$$\tilde{Y}(.)\Phi \triangleq M(q)\ddot{q} + C(q,\dot{q})(\dot{q}_d + \lambda e) + G(q) - M(q_d)\ddot{q}_d - C(q_d,\dot{q}_d)\dot{q}_d - G(q_d) \tag{A.3}$$

Applying the control input of the proposed extended DCAL controller (3.11) to the error dynamics in (B.4) results in

$$M(q)\dot{e}_v = -C(q,\dot{q})e_v - K_P(.)e - K_V(.)e_v - \sigma\|e\|^2 e_v + \tilde{Y}(.)\Phi + Y_d(.)\tilde{\Phi}(t) \tag{A.4}$$

with $\tilde{\Phi}(t) \triangleq \Phi - \hat{\Phi}(t)$ is the estimation error vector and $Y_d \triangleq Y(q_d, \dot{q}_d, \ddot{q}_d)$.

For the stability analysis of the system's error dynamics given by (B.5), consider the following Lyapunov candidate

$$V = \frac{1}{2} e_v^T M(q) e_v + \frac{1}{2} e^T K_P(.) e + \frac{1}{2} \tilde{\Phi}^T K^{-1} \tilde{\Phi} \tag{A.5}$$

whose first time-derivative leads to

$$\dot{V} = \frac{1}{2} e_v^T \dot{M}(q) e_v + e_v^T M(q) \dot{e}_v + e^T K_P(.) \dot{e} + \tilde{\Phi}^T K^{-1} \dot{\tilde{\Phi}} \tag{A.6}$$

Using the skew-symmetric property of $\dot{M}(q) - 2C(q, \dot{q})$ and the adaptation law (3.4), $\dot{V}$ can be rewritten as follows

$$\dot{V} = -e^T K_P(.) e - e_v^T K_V(.) e_v - \sigma \|e\|^2 \|e_v\|^2 + e_v^T \tilde{Y}(.) \Phi \tag{A.7}$$

Following the same reasoning in [Sadegh and Horowitz, 1990], $\tilde{Y}(.)\Phi$ can be upper-bounded as follows

$$\left\| \tilde{Y}(.) \Phi \right\| \leq \zeta_1 \|e\| + \zeta_2 \|e\|^2 + \zeta_3 \|e_v\| + \zeta_4 \|e_v\| \|e\| \tag{A.8}$$

where $\zeta_1, \zeta_2, \zeta_3$ and ζ_4 are positive bounding constants. If we consider the upper bound on $\left\| \tilde{Y}(.) \Phi \right\|$ together with the upper bound on $K_P(.)$ and $K_V(.)$, $\dot{V}$ can be upper-bounded as follows

$$\dot{V} \leq -K_{P_M} \|e\|^2 - K_{V_M} \|e\|^2 - \sigma \|e\|^2 \|e_v\|^2 + \zeta_1 \|e\| \|e_v\| + \zeta_2 \|e\|^2 \|e_v\| + \zeta_3 \|e_v\|^2 + \zeta_4 \|e_v\|^2 \|e\| \tag{A.9}$$

Rearranging the terms of (A.9), it can be rewritten as

$$\begin{aligned} \dot{V} \leq & -K_{P_M} \|e\|^2 - K_{V_M} \|e\|^2 - \sigma \|e\|^2 \|e_v\|^2 + \zeta_1 \|e\| \|e_v\| + \zeta_2 \|e\|^2 \left[\frac{1}{2} - \|e_v\|\right]^2 \\ & + \zeta_4 \|e_v\|^2 \left[\frac{1}{2} - \|e\|\right]^2 + (\zeta_2 + \zeta_4) \|e\|^2 \|e_v\|^2 + \frac{\zeta_2}{4} \|e\|^2 + (\zeta_3 + \frac{\zeta_4}{4}) \|e_v\|^2 \end{aligned} \tag{A.10}$$

Then, by collecting common terms of (A.10), we obtain

$$\begin{aligned} \dot{V} \leq & -(K_{P_M} - \frac{\zeta_2}{4}) \|e\|^2 - (K_{V_M} - \zeta_3 - \frac{\zeta_4}{4}) \|e_v\|^2 + \zeta_1 \|e\| \|e_v\| \\ & - \zeta_2 \|e\|^2 \left[\frac{1}{2} - \|e_v\|\right]^2 - \zeta_4 \|e_v\|^2 \left[\frac{1}{2} - \|e\|\right]^2 - (\sigma - \zeta_2 - \zeta_4) \|e\|^2 \|e_v\|^2 \end{aligned} \tag{A.11}$$

If the control gain σ is chosen such that $(\sigma - \zeta_2 - \zeta_4) > 0$, then, $\dot{V}$ can be upper-bounded with the following new bound

$$\dot{V} \leq -(K_{P_M} - \frac{\zeta_2}{4}) \|e\|^2 - (K_{V_M} - \zeta_3 - \frac{\zeta_4}{4}) \|e_v\|^2 + \zeta_1 \|e\| \|e_v\| \tag{A.12}$$

Then, considering the fact that

$$\|e\|\,\|e_v\| \leq \frac{\|e\|^2}{2} + \frac{\|e_v\|^2}{2} \tag{A.13}$$

leads to the new upper-bound on $\dot{V}$ as follows

$$\dot{V} \leq -(K_{P_M} - \frac{\zeta_2}{4} - \frac{\zeta_1}{2})\,\|e\|^2 - (K_{V_M} - \zeta_3 - \frac{\zeta_4}{4} - \frac{\zeta_1}{2})\,\|e_v\|^2 \tag{A.14}$$

Consequently, if the upper-bounds K_{P_M} and K_{V_M} are tuned such that

$$K_{P_M} - \frac{\zeta_2}{4} - \frac{\zeta_1}{2} > 0 \tag{A.15}$$

$$K_{V_M} - \zeta_3 - \frac{\zeta_4}{4} - \frac{\zeta_1}{2} > 0 \tag{A.16}$$

then, $\dot{V}$ would be negative semi-definite. Considering the bound inequalities of $K_P(.)$ and $K_V(.)$ in (3.14) and (3.15), respectively, we can notice that if the gains parameters are chosen such that

$$k_{p_0}\delta_p^{\alpha_p - 1} > \frac{\zeta_2}{4} + \frac{\zeta_1}{2} \tag{A.17}$$

$$k_{v_0}\delta_v^{\alpha_v - 1} > \zeta_3 + \frac{\zeta_4}{4} + \frac{\zeta_1}{2} \tag{A.18}$$

then, the inequalities (A.15) and (A.16) are satisfied. Consequently, $\dot{V}$ is negative semi-definite, therefore, V is bounded. Then, based on the definition of V, we conclude that $e, \dot{e}, r$ and $\tilde{\Phi}$ are all bounded. Eq. (B.5) can be used to show that $\dot{r}(t)$, $\ddot{q}(t)$ and, consequently, $\ddot{V}$ are all bounded. Given the fact that the inertia matrix $M(q)$ is lower bounded, it can be noticed that V in (A.5) is lower bounded as well. Since V is lower bounded, $\dot{V}$ is negative semi-definite and $\ddot{V}$ is bounded (i.e. $\dot{V}$ is uniformally continuous). Finally, based on Barbalat's Lemma [Khalil and Grizzle, 1996], we can state that

$$\lim_{t\to\infty} \dot{V} = 0 \tag{A.19}$$

Therefore, from the definition of $\dot{V}$ and r, we conclude that

$$\lim_{t\to\infty} e = 0 \tag{A.20}$$

$$\lim_{t\to\infty} \dot{e} = 0 \tag{A.21}$$

which concludes the proof. Notice that the stability analysis does not say anything about the estimated parameters except that they remain bounded.

□

APPENDIX

B

Stability analysis of the RISE-based adaptive controller

Theorem B.1 *The joint position tracking error* $e_1(t)$ *a PKM whose dynamics is governed by* (3.23) *under the controller* (3.30)*, with* $\Gamma_d(t) = 0$*, along with the adaptation law* (3.31) *goes to zero as time goes to infinity, provided that the control design parameters* α_1 *and* α_2 *are chosen such that* $\alpha_1 > 1/2$*,* $\alpha_2 > 1$*,* $\beta > \zeta_{N_d} + (\zeta_{N_{d2}}/\alpha_2)$ *and that the feedback gain* k_s *is chosen large enough.*

Proof. Substituting the proposed control law (3.30) into the dynamics of the PKM (3.23) results in the following closed-loop error system:

$$M(q)\nu = Y_d(\cdot)\tilde{\Phi}(t) + W(\cdot) - \Gamma(t), \tag{B.1}$$

where

$$\nu(t) \triangleq \dot{r}(t) + \alpha_2 r(t) \tag{B.2}$$

is an auxiliary unmeasurable error signal, $\tilde{\Phi}(t) \triangleq \Phi - \hat{\Phi}(t)$ is the parameters' estimation error and the nonlinear function $W(\cdot)$ is given by

$$W(\cdot) = M(q)\left(\ddot{q}_d + \alpha_1 \dot{e}_1 + \alpha_2 e_2\right) + C(q,\dot{q})\dot{q} + G(q) - Y_d(\cdot)\Phi. \tag{B.3}$$

The first time-derivative of (B.1) is expressed by

$$M(q)\dot{\nu} = -\dot{M}(q)\nu + \dot{Y}_d(\cdot)\tilde{\Phi}(t) + Y_d(\cdot)\dot{\tilde{\Phi}}(t) + \dot{W}(\cdot) - \dot{\Gamma}(t) \tag{B.4}$$

After rearranging the terms in (B.4), it can be rewritten as follows:

$$M(q)\dot{\nu} = -\frac{1}{2}\dot{M}(q)\nu + \dot{Y}_d(\cdot)\tilde{\Phi}(t) + N(\cdot) - \dot{\Gamma}(t) - e_2 \tag{B.5}$$

where the nonlinear function $N(\cdot)$ is given by:

$$N(\cdot) = -Y_d(\cdot)\dot{\hat{\Phi}}(t) + \dot{W}(\cdot) + e_2 - \frac{1}{2}\dot{M}(q)v \tag{B.6}$$

Now, consider another auxiliary nonlinear function $N_d(\cdot)$ obtained by replacing all occurrences of $q, \dot{q}$ and $\ddot{q}$ in (B.6) by $q_d, \dot{q}_d$ and $\ddot{q}_d$, respectively. Adding and subtracting $N_d(\cdot)$ to the error dynamics (B.5) yields:

$$M(q)\dot{v} = -\frac{1}{2}\dot{M}(q)v + \dot{Y}_d(\cdot)\tilde{\Phi}(t) - \dot{\Gamma}(t) - e_2 + \tilde{N}(\cdot) + N_d(\cdot), \tag{B.7}$$

where $\tilde{N}(\cdot) \triangleq N(\cdot) - N_d(\cdot)$. In a similar manner to [Patre et al., 2008], $\tilde{N}(\cdot)$ can be upper-bounded as follows

$$\left\|\tilde{N}(\cdot)\right\| \leq \rho\left(\|z\|\right)\|z\|, \tag{B.8}$$

where $z(t) \triangleq [e_1^T e_2^T v^T]^T$ and $\rho(\cdot)$ is some nondecreasing positive invertible function.

Based on the definitions of $N_d(\cdot)$ and $Y_d(\cdot)$, the following upper-bounds can be developed

$$\left\|N_d(\cdot)\right\|_\infty \leq \zeta_{N_d}, \qquad \left\|\dot{N}_d(\cdot)\right\|_\infty \leq \zeta_{N_{d2}} \tag{B.9}$$

$$\left\|Y_d(\cdot)\right\|_\infty \leq \zeta_{Y_d}, \qquad \left\|\dot{Y}_d(\cdot)\right\|_\infty \leq \zeta_{Y_{d2}} \tag{B.10}$$

Let $\mathcal{D} \subset \mathbb{R}^{3n+p+1}$ be a domain containing $y(t) \triangleq [z^T(t)\ \tilde{\Phi}^T(t)\sqrt{P(t)}]^T = 0$, where $P(t) \in \mathbb{R}$ is an auxiliary function defined as

$$P(t) \triangleq \beta\left\|e_2(t_0)\right\| - e_2^T(t_0)N_d(t_0) - \int_{t_0}^{t} L(\sigma)d\sigma, \tag{B.11}$$

being $L(t) \triangleq v^T\left(N_d(t) - \beta\,\mathrm{sgn}(e_2)\right) \in \mathbb{R}$. Provided that the control parameter β is chosen such that $\beta > \zeta_{N_d} + (\zeta_{N_{d2}}/\alpha_2)$, the following inequality holds:

$$\int_{t_0}^{t} L(\sigma)d\sigma \leq \beta\left\|e_2(t_0)\right\| - e_2^T(t_0)N_d(t_0), \tag{B.12}$$

meaning that $P(t) \geq 0,\ \forall t \geq t_0$.

Now, let $V(y,t) : \mathcal{D} \times [0,\infty) \to \mathbb{R}$ be a continuously differentiable positive definite function given by

$$V(y,t) = e_1^T e_1 + \frac{1}{2}e_2^T e_2 + \frac{1}{2}v^T M(q)v + P + \frac{1}{2}\tilde{\Phi}\Lambda^{-1}\tilde{\Phi}, \tag{B.13}$$

satisfying the following bounding inequality:

$$\eta_1\|y\|^2 \leq V(y,t) \leq \eta_2(q)\|y\|^2, \tag{B.14}$$

being

$$\eta_1 \triangleq \frac{1}{2}\min\left\{1, \underline{m}, \lambda_{\min}\left\{\Gamma^{-1}\right\}\right\} \tag{B.15}$$

$$\eta_2(q) \triangleq \max\left\{1, \frac{1}{2}\bar{m}(q), \frac{1}{2}\lambda_{\max}\left\{\Gamma^{-1}\right\}\right\}, \tag{B.16}$$

for some known positive constant $\underline{m}$ and some positive nondecreasing function $\bar{m}(q)$ such that:

$$\underline{m}\,\|y\|^2 \leq y^T M(q) y \leq \bar{m}(q)\,\|y\|^2, \tag{B.17}$$

and $\lambda_{\min}\{\cdot\}$ and $\lambda_{\max}\{\cdot\}$ are the minimum and maximum eigenvalue of their argument, respectively. Differentiating (B.13) with respect to time yields

$$\dot{V}(y,t) = \nu^T M(q)\dot{\nu} + \frac{1}{2}\nu^T \dot{M}(q)\nu + e_2^T \dot{e}_2 + 2e_1^T \dot{e}_1 + \dot{P} - \tilde{\Phi}^T \Lambda^{-1}\dot{\hat{\Phi}}. \tag{B.18}$$

Upon the use of (3.26), (B.2), (B.7) and the first time-derivatives of (3.27) and (B.11), Eq. (B.18) can be rewritten as follows:

$$\dot{V}(y,t) = \nu^T \tilde{N}(t) - (k_s+1)\,\|\nu\|^2 - \alpha_2\,\|e_2\|^2 - 2\alpha_1\,\|e_1\|^2 + 2e_2^T e_1 + \nu^T \dot{Y}_d(\cdot)\tilde{\Phi} - \tilde{\Phi}^T Y_d(\cdot) e_2. \tag{B.19}$$

Considering the fact that $e_2^T e_1 \leq (\|e_1\|^2 + \|e_2\|^2)/2$ and given that $\tilde{\Phi} \leq \Phi + \Phi_b$, upon the use of (B.10), $\dot{V}(\cdot)$ can be upper-bounded as follows:

$$\dot{V}(y,t) \leq \nu^T\left(\tilde{N}(t) + \zeta_{Y_{d2}}(\Phi + \Phi_b)\right) - (k_s+1)\,\|\nu\|^2 - \alpha_2\,\|e_2\|^2 - 2\alpha_1\,\|e_1\|^2 + \|e_1\|^2 + \|e_2\|^2 - (\Phi+\Phi_b)^T \zeta_{Y_d}\,\|e_2\|. \tag{B.20}$$

Given that $(\Phi+\Phi_b)^T \zeta_{Y_d}\,\|e_2\| \geq 0$ and taking (B.8) into account, the new following upper-bound on $\dot{V}(\cdot)$ can be obtained:

$$\dot{V}(y,t) \leq \rho_1(\|z\|)\,\|z\|\,\|e_2\| - (k_s+1)\,\|\nu\|^2 - \alpha_2\,\|e_2\|^2 - 2\alpha_1\,\|e_1\|^2 + \|e_1\|^2 + \|e_2\|^2, \tag{B.21}$$

where $\rho_1(\|z\|) \triangleq \rho(\|z\|) + \zeta_{Y_{d2}}(\Phi + \Phi_b)$. After rearranging the terms, (B.21) can be rewritten as follows:

$$\dot{V}(y,t) \leq -\eta_3\,\|z\|^2 - \left(k_s\,\|\nu\|^2 - \rho_1(\|z\|)\,\|\nu\|\,\|z\|\right), \tag{B.22}$$

where $\eta_3 \triangleq \min\{2\alpha_1 - 1, \alpha_2 - 1, 1\}$, meaning that the α_1 and α_2 should be chosen such that:

$$\alpha_1 > \frac{1}{2}, \qquad \alpha_2 > 1.$$

By completing the squares of the last term of (B.22), the following upper-bound of $\dot{V}(\cdot)$ can be obtained:

$$\dot{V}(y,t) \leq -\eta_3 \|z\|^2 + \frac{\rho_1^2(z)\|z\|^2}{4k_s} \triangleq c\|z\|^2, \tag{B.23}$$

for some positive constant $c \in \mathbb{R}$. In (B.23), the function $c\|z\|^2$ is a continuous positive semi-definite function defined on:

$$D \triangleq \left\{ y \in \mathbb{R}^{3n+p+1} \,|\, \|y\| \leq \rho_1^{-1}\left(2\sqrt{\eta_3 k_s}\right) \right\}.$$

Let $S \subset \mathcal{D}$ be the set defined by:

$$S \triangleq \left\{ y(t) \subset \mathcal{D} | c\|z\|^2 < \eta_1 \left(\rho_1^{-1}\left(2\sqrt{\eta_3 k_s}\right)\right)^2 \right\}. \tag{B.24}$$

Following the same reasoning in [**?**], we can conclude that $c\|z\|^2$ is uniformly continuous in $\mathcal{D}$ (due to the boundedness of $\dot{e}, \dot{r}$ and $\dot{v}$ in $\mathcal{D}$). Invoking Theorem 8.4 of [Khalil and Grizzle, 1996], it can be concluded that:

$$c\|z\|^2 \to 0 \qquad \text{as} \qquad t \to \infty \qquad \forall y(t_0) \in S, \tag{B.25}$$

which means, based on the definition of $z(t)$, that:

$$\|e(t)\| \to 0 \qquad \text{as} \qquad t \to \infty \qquad \forall y(t_0) \in S, \tag{B.26}$$

which concludes the proof.

□

APPENDIX

C Dynamic modeling of experimental platforms

C.1 Delta robot

The dynamic model of the Delta robot can be obtained using the virtual work principle as described in [Codourey, 1998]. This principle states that the contribution of all non-inertial forces must equal the contribution of inertial ones.

Starting with the dynamics of the traveling plate, two kinds of forces act on it. The force due to gravity $G_{tp} \in \mathbb{R}^3$ and the force due to its acceleration $F_{tp} \in \mathbb{R}^3$ (inertial force). They can be expressed in operational coordinates as follows

$$G_{tp} = m_{tp}[0\ 0\ -g]^T \tag{C.1}$$

$$F_{tp} = m_{tp}\ddot{X} \tag{C.2}$$

being m_{tp} the mass of the traveling plate in addition to the half-masses of the forearms, $g = 9.81\ m/s^2$ is the gravity constant and $\ddot{X} \in \mathbb{R}^3$ is the acceleration vector of the traveling-plate.

These forces, expressed in the traveling-plate coordinates, can be mapped to the actuators' coordinates by using the Jacobian matrix $J \in \mathbb{R}^{3\times3}$ to obtain their contribution to each motor as follows

$$\Gamma_{G_{tp}} = J^T m_{tp}[0\ 0\ -g]^T \tag{C.3}$$

$$\Gamma_{tp} = J^T m_{tp}\ddot{X} \tag{C.4}$$

At the actuators' level, the forces that are considered are the actuators' torques, the forces due to the rear-arms rotations and the gravity forces acting on the rear-arms.

The virtual work principle states that the contribution of all non-inertial forces must equal the contribution of all inertial forces. Applying this statement on the actuators' level

we get

$$\Gamma + J^T G_{tp} + \Gamma_{G_a} = I_a \ddot{q} + J^T m_{tp} \ddot{X} \tag{C.5}$$

with $\Gamma \in \mathbb{R}^3$ the vector of torques of the actuators, $I_a \in \mathbb{R}^{3\times3}$ is a diagonal inertia matrix that includes the inertia of the rear-arms and the half-masses of the forearms with respect to the actuators' rotation axes. $\Gamma_{G_a} \in \mathbb{R}^3$ is the torque due to the gravitational forces of the arms which is given by

$$\Gamma_{G_a} = m_a r_{G_a} g \cos\left[\cos(q_1)\cos(q_2)\cos(q_3)\right]^T \tag{C.6}$$

with m_a being the mass of one rear-arm and one half-mass of one forearm, r_{G_a} the position of the center of mass of the arm and q_1, q_2 and q_3 are the joints positions.

Rearranging the terms of (C.5), we can write

$$I_a \ddot{q} + J^T m_{tp} \ddot{X} - J^T G_{tp} - \Gamma_{G_a} = \Gamma \tag{C.7}$$

The joint acceleration vector and the traveling-plate vector are in fact linked by the following kinematic relationship

$$\ddot{X} = J\ddot{q} + \dot{J}\dot{q} \tag{C.8}$$

with $\dot{q} \in \mathbb{R}^3$ being the joint velocity vector. Substituting (C.8) into C.7 yields

$$I_a \ddot{q} + J^T m_{tp} \left(J\ddot{q} + \dot{J}\dot{q}\right) - J^T G_{tp} - \Gamma_{G_a} = \Gamma \tag{C.9}$$

Rearranging the terms of C.9 we obtain

$$\left(I_a + m_{tp} J^T J\right)\ddot{q} + m_{tp} J^T \dot{J}\dot{q} - J^T G_{tp} - \Gamma_{G_a} = \Gamma \tag{C.10}$$

which can be written into the standard joint-space formulation as follows

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \Gamma \tag{C.11}$$

being

$M(q) = I_a + m_{tp} J^T J$ the total mass matrix of the manipulator.

$C(q,\dot{q}) = m_{tp} J^T \dot{J}$ the Coriolis and centrifugal forces matrix.

$G(q) = -J^T G_{tp} - \Gamma_{G_a}$ the gravity forces vector.

C.2 VELOCE robot

The dynamics of the VELOCE robot shows a lot of similarities with those of the Delta robot. Nevertheless few differences arise due to the number of kinematic chains and the additional rotational DOF of the traveling plate.

Regarding the traveling plates, we distinguish two kinds of forces acting on it, the gravity forces $G_{tp} \in \mathbb{R}^4$ and the inertial force $F_{tp} \in \mathbb{R}^4$, they are given by

$$G_{tp} = M_{tp} [0\ 0\ -g\ 0]^T \tag{C.12}$$

$$F_{tp} = M_{tp} \ddot{X} \tag{C.13}$$

where $M_{tp} \in \mathbb{R}^{4\times4}$ is the mass matrix of the traveling-plate that also considers its rotational movement as well as the half-masses of the four forearms, $g = 9.81\ m/s^2$ is the gravity constant and $\ddot{X} \in \mathbb{R}^4$ is the Cartesian acceleration vector.

The contributions of G_{tp} and F_{tp} to each motor can be computed using the Jacobian matrix as follows

$$\Gamma_{G_{tp}} = J^T M_{tp} [0\ 0\ -g\ 0]^T \tag{C.14}$$

$$\Gamma_{tp} = J^T M_{tp} \ddot{X} \tag{C.15}$$

From the joints side, the elements that contribute to the dynamics of the actuators are the forces and torques resulting from the movement of the rear-arms in addition to the half-messes of the forearms.

Applying the virtual work principle, which states that the sum of non-inertial forces is equal to that of the inertial ones, and after rearranging the terms, we get

$$J^T M_{tp} \ddot{X} + \Gamma_{tp} + I_a \ddot{q} + \Gamma_{G_a} = \Gamma \tag{C.16}$$

being $I_a \in \mathbb{R}^{4\times4}$ a diagonal inertia matrix of the arms accounting for the rear-arms as well as the half-masses of the forearms, $\ddot{q} \in \mathbb{R}^4$ the joint acceleration vector and $\Gamma_{G_a} \in \mathbb{R}^4$ is the force vector resulting from gravity acting on the arms being given by

$$\Gamma_{G_a} = m_a r_{G_a} g \cos [\cos(q_1)\ \cos(q_2)\ \cos(q_3)\ \cos(q_4)]^T \tag{C.17}$$

with m_a the sum of the mass of one rear-arm and the half-mass of one forearm, r_{G_a} the distance between the center of one axis and the center of mass of one arm and q_1, q_2, q_3 and q_4 are the joints positions.

Given the kinematic relationship $\ddot{X} = J\ddot{q} + \dot{J}\dot{q}$, equation C.16 can be rewritten as follows

$$(I_a + J^T M_{tp} J)\ \ddot{q} + (J^T M_{tp} \dot{J})\ \dot{q} - (J^T G_{tp} + \Gamma_{G_a}) = \Gamma \tag{C.18}$$

which is in a standard joint-space form similar to C.11 with

$M(q) = I_a + J^T M_{tp} J$ the total mass matrix.

$C(q, \dot{q}) = J^T M_{tp} \dot{J}$ the Coriolis and centrifugal forces matrix.

$G(q) = -J^T G_{tp} - \Gamma_{G_a}$ the gravitational forces vector.

C.3 DUAL-V robot

The DUAL-V dynamic modeling technique is quite similar to Delta-like robots. Nevertheless, the simplifying hypotheses of neglecting the rotational inertia of the forearms is no longer valid. The reason behind this is the use of relatively heavy material for the forearms. Then, an additional term has to be added to compensate for the additional inertia of the two half-masses model of the forearms as it will be explained shortly.

Since the DUAL-V robot has only planar motions, gravity effects are non existent. Hence, from the actuators side we can write the following equation

$$\Gamma = I_b \ddot{q} \tag{C.19}$$

where $I_b \in \mathbb{R}^{4\times4}$ is the inertia matrix of the arms that accounts for the motion of the arms, the actuators, the counter-masses and the half-masses of the forearms. $\Gamma \in \mathbb{R}^4$ and $\ddot{q} \in \mathbb{R}^4$ are the actuators' torque and the joint acceleration vectors respectively.

The only forces acting on the traveling plate are the ones emerging from its inertia F_{tp} which are given by

$$F_{tp} = M_{tp}\ddot{X} \tag{C.20}$$

where $M_{tp} \in \mathbb{R}^{3\times3}$ is the mass matrix of the traveling-plate, including also the half-masses of the forearms. $\ddot{X} \in \mathbb{R}^3$ is the acceleration vector of the traveling-plate which also includes its rotational movement.

To compute the contribution of the F_{tp} to the actuators' dynamics, the inverse Jacobian matrix $J_m \in \mathbb{R}^{4\times3}$ is used. However, one consideration should be taken into account. Indeed, unlike non-redundantly actuated PKMs, the inverse Jacobian matrix in the case of actuation redundancy is not square and, hence, cannot be inverted. This can be explained by the existence of unlimited set of actuators' torques Γ that can counter balance a specific force acting on the traveling plate.

A solution that is widely adopted in robotics is the use of the pseudoinverse of the inverse Jacobian matrix. This solution minimizes the Euclidean norm of the actuators' torque vector. Consequently, we can write the contribution of the F_{tp} to each actuator as follows

$$\Gamma_{tp} = J_m^{T*} M_{tp}\ddot{X} \tag{C.21}$$

with $J_m^{T*} \in \mathbb{R}^{4\times3}$ denoting the pseudoinverse of J_m^T. Equivalently, the pseudoinverse of J_m can be used to express the relationship between the joint and Cartesian acceleration vectors according to the following equation

$$\ddot{X} = J_m^* \ddot{q} - J_m^* \dot{J}_m J_m \dot{q} \tag{C.22}$$

Substituting (C.22) into yields

$$\Gamma_{tp} = J_m^{T*} M_{tp} \left(J_m^* \ddot{q} - J_m^* \dot{J}_m J_m \dot{q} \right) \tag{C.23}$$

Since the forearms are considered as two point-masses located at their ends, their contribution is considered in (C.19) and (C.20). In addition, a specific inertia which corresponds to the two-masses model is also included [van der Wijk et al., 2013]. Consequently, an additional term that subtracts the modeled inertia from the real one (obtained from CAD software for instance) is added. This term compensates for the modeling error emerging from the simplifying forearms hypotheses which may have significant impact on the accuracy of the dynamic model in the case of the DUAL-V robot. We denote this additional term by $\Gamma_{COMP} \in \mathbb{R}^4$ and, for compactness, is not explained herein.

At last, the full inverse dynamic model of the DUAL-V robot is expressed by

$$\left(I_b + J_m^{T*} M_{tp} J_m^* \right) \ddot{q} - \left(J_m^{T*} M_{tp} J_m^* \dot{J}_m J_m \right) \dot{q} + \Gamma_{comp} = \Gamma \tag{C.24}$$

For more details, the reader may refer to [van der Wijk et al., 2013].

C.4 ARROW robot

To establish the dynamic model of the ARROW robot, the same hypotheses usually made for Delta-like robots are assumed valid. These hypotheses can be summarized in the following two points

Dry and viscous friction forces are neglected in all passive and active joints.

The mass of the forearms is split into two point masses located at both ends of the arms. Hence their contribution to the dynamics are considered with the linear actuators and the traveling plate.

Applying the law of motion, the dynamics of the linear actuators are given by the following equation

$$M_a \ddot{q} = \Gamma - J_q^T f \tag{C.25}$$

being $M_a = \text{diag}(m_{as}, m_{as}, m_{ap}, m_{ap}, m_{as}, m_{as})$, where m_{as} includes the mass of the actuator, the moving cart and the half-mass of the simple arm. m_{ap} includes the mass of the actuator, the moving cart and the half-mass of the parallelogram (i.e. two simple arms). $\ddot{q} \in \mathbb{R}^6$ is the joint acceleration vector, $\Gamma \in \mathbb{R}^6$ is the input force vector, $J_q \in \mathbb{R}^{6\times6}$ is the joint Jacobian matrix and $f \in \mathbb{R}^6$ is the force vector resulting from acceleration and gravitational forces acting on the moving platform.

From the platform side, the dynamics are governed by

$$M_p \ddot{X} + \Lambda_c \dot{X} = J_x^T f + m_p G \tag{C.26}$$

where $M_p \in \mathbb{R}^{4\times4}$ is the mass matrix of the moving platform, $\ddot{X} \in \mathbb{R}^4$ is its acceleration vector, $\Lambda_c \in \mathbb{R}^{4\times4}$ is the matrix of Coriolis and centrifugal effects of the platform, $\dot{X} \in \mathbb{R}^4$ is the platform velocity vector, $J_x \in \mathbb{R}^{6\times4}$ is the Cartesian Jacobian matrix, m_p is the total mass of the platform including the half-masses from the arms and parallelograms and $G = [0,0,-9.8,0]^T m/s^2$ is the gravity wrench acting on the platform. Merging (C.25) and (C.26) and substituting $\ddot{q} = J_m\ddot{X} + \dot{J}_m\dot{X}$, with $J_m = J_q^{-1}J_x$ the inverse Jacobian matrix, yields the following equation

$$\ddot{X} = H\Gamma - \Lambda\dot{X} + A_G \tag{C.27}$$

where $H \in \mathbb{R}^{4\times6}$, $\Lambda \in \mathbb{R}^{4\times4}$ and $A_G \in \mathbb{R}^4$ are expressed by

$$\begin{cases} H = (M_p + J_m^T M_a J_m)^{-1} J_m^T \\ \Lambda = H M_a \dot{J}_m + (M_p + J_m^T M_a J_m)^{-1} \Lambda_c \\ A_G = (M_p + J_m^T M_a J_m)^{-1} m_p G \end{cases} \tag{C.28}$$

The inverse dynamic model (IDM) is not unique due to actuation redundancy and is obtained by using the pseudo-inverse of the matrix H as follows

$$H^* (\ddot{X} + \Lambda\dot{X} - A_G) = \Gamma \tag{C.29}$$

where H^* denotes the pseudo-inverse of H. For more details about the kinematic and dynamic modeling of ARROW robot, the reader is referred to [Shayya, 2015].

APPENDIX

D

Interpolation points used in experiments

We have mentioned that the trajectories used in real-time experiments performed at LIRMM are generated according to the method described in [Khalil and Dombre, 2004]. This method, described in 5.4, consists of generating a point-to-point trajectory for the end-effector of the traveling plate between a starting and ending point according to a fifth order polynomial. The motion starts from the initial point X_i with zero velocity and acceleration ($\dot{X}_d = \ddot{X}_d = 0$) and ends at a final point X_f also with zero velocity and acceleration ($\dot{X}_f = \ddot{X}_f = 0$). Then, the actual final point becomes the initial point for the next motion, and a new target final position is specified, and so on.

In what follows, the sequence of Cartesian coordinates for each experienced control scheme is provided (except for the Delta robot which uses semi-elliptic 3D pick-and-place trajectories).

Table D.1 – Contribution 1: Sequence of points for the interpolated trajectory

Point	x [mm]	y [mm]	α[deg]	T [s]
X_0	0	0	0	
X_1	40	100	0	0.25
X_2	0	0	0	0.25
X_3	-40	-100	0	0.25
X_4	0	0	0	0.25
X_5	0	0	10	0.25
X_6	0	0	0	0.25
X_7	0	0	-10	0.25
X_8	0	0	0	0.25

Table D.2 – Contribution 2: Sequence of points for the interpolated trajectory

Point	x [mm]	y [mm]	α[deg]	T [s]
X_0	0	0	0	
X_1	20	40	15	0.4
X_2	0	0	0	0.4
X_3	-20	-40	-15	0.4
X_4	0	0	0	0.4
X_5	20	-40	15	0.4
X_6	0	0	0	0.4
X_7	-20	40	-10	0.4
X_8	0	0	0	0.4

Table D.3 – Contribution 3 and 4: Sequence of points for the interpolated trajectory

Point	x [mm]	y [mm]	z [mm]	α[deg]	T [s]
X_0	0	0	0	0	
X_1	150	150	0	-30	0.2
X_2	-150	150	0	-30	0.2
X_3	-150	-150	0	-30	0.2
X_4	150	-150	0	-30	0.2
X_5	150	150	0	-30	0.2
X_6	-150	150	0	-30	0.2
X_7	-150	-150	0	-30	0.2
X_8	150	-150	0	-30	0.2
X_9	150	150	0	-30	0.2
X_{10}	-150	-150	0	-30	0.2
X_{11}	-150	-150	0	-30	0.2
X_{12}	150	-150	0	-30	0.2
X_{13}	0	0	0	0	0.2
X_{14}	0	0	50	0	0.2
X_{15}	0	0	-100	0	0.2
X_{16}	0	0	50	0	0.2
X_{17}	0	0	-100	0	0.2
X_{18}	0	0	0	0	0.2
X_{19}	0	0	0	-30	0.2
X_{20}	0	0	0	30	0.2
X_{21}	0	0	0	-30	0.2
X_{22}	0	0	0	30	0.2
X_{23}	0	0	0	0	0.2

Table D.4 – Contribution 5: Sequence of points for the interpolated trajectory

Point	x [mm]	y [mm]	z [mm]	α[deg]	T [s]
X_0	0	0	0	0	
X_1	100	100	-150	0	0.7
X_2	-100	100	-150	0	0.5
X_3	-100	100	50	0	0.5
X_4	100	100	50	0	0.7
X_5	100	100	-150	0	0.5
X_6	100	-100	-150	0	0.7
X_7	-100	-100	-150	0	0.5
X_8	-100	-100	50	0	0.5
X_9	100	-100	50	0	0.5
X_{10}	100	-100	-150	0	0.5
X_{11}	0	0	0	0	0.7
X_{12}	0	0	0	45	0.5
X_{13}	0	0	0	-45	0.7
X_{14}	0	0	0	45	0.7
X_{15}	0	0	0	-45	0.5
X_{16}	0	100	0	0	0.7
X_{17}	0	100	-150	45	0.5
X_{18}	0	100	-150	-45	0.7
X_{19}	0	100	-150	45	0.5
X_{20}	0	100	-150	-45	0.5
X_{21}	0	100	-150	0	0.7
X_{22}	0	100	50	0	0.5
X_{23}	0	100	50	45	0.5
X_{24}	0	100	50	-45	0.7
X_{25}	0	100	50	45	0.7
X_{26}	0	100	50	-45	0.7
X_{27}	0	0	0	0	0.5
X_{28}	200	0	-150	0	0.7
X_{29}	-200	0	-150	0	0.7
X_{30}	200	0	-150	0	0.7
X_{31}	0	0	0	0	0.5

Bibliography

H. Abdellatif and B. Heimann. Computational efficient inverse dynamics of 6-dof fully parallel manipulators by using the lagrangian formalism. *Elsevier Mechanism and Machine Theory*, 44(1):192–207, 2009. Cited page 32.

B.D.O. Anderson. Failures of adaptive control theory and their resolution. *Communications in Information & Systems*, 5(1):1–20, 2005. Cited page 73.

J. Angeles. *Fundamentals of robotic mechanical systems*. Springer, 2002. Cited page 8.

M. Asgari and M.A. Ardestani. Dynamics and improved computed torque control of a novel medical parallel manipulator: Applied to chest compressions to assist in cardiopulmonary resuscitation. *Journal of Mechanics in Medicine and Biology*, 15(04):1550051, 2015. Cited page 42.

K.J. Astrom. *Adaptive Control (Second Edition)*. Addison-Wesley, 1995. Cited page 46.

M. Becerra-Vargas and E.M. Belo. Application of h-infinity theory to a 6 dof flight simulator motion base. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 34 (2):193–204, 2012. Cited page 45.

P. Begon, F. Pierrot, and P. Dauchez. Fuzzy sliding mode control of a fast parallel robot. In *IEEE International Conference on Robotics and Automation (ICRA'95)*, volume 1, pages 1178–1183, Nagoya, Japan, May 1995. Cited page 40.

M. Bennehar, A. Chemori, and F. Pierrot. A new extension of direct compensation adaptive control and its real-time application to redundantly actuated pkms. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'14)*, pages 1670–1675, Chicago, Illinois, USA, September 2014a. Cited page 59.

M. Bennehar, A. Chemori, and F. Pierrot. A novel rise-based adaptive feedforward controller for redundantly actuated parallel manipulators. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'14)*, pages 2389–2394, Chicago, Illinois, USA, September 2014b. Cited page 68.

M. Bennehar, A. Chemori, F. Pierrot, , and V. Creuze. Extended model-based feedforward compensation in $\mathcal{L}_1$ adaptive control for mechanical manipulators: Design and experiments. *Frontiers in Robotics and AI*, 2015a. Submitted. Cited page 85.

M. Bennehar, A. Chemori, and F. Pierrot. A new revised desired compensation adaptive control for enhanced tracking: Application to ra-pkms. *Advanced Robotics*, 2015b. Submitted. Cited page 59.

M. Bennehar, A. Chemori, and F. Pierrot. Feedforward augmented $\mathcal{L}_1$ adaptive controller for parallel kinematic manipulators with improved tracking. *IEEE Robotics and Automation Letters*, 2015c. Submitted. Cited page 82.

M. Bennehar, A. Chemori, and F. Pierrot. L1 adaptive control of parallel kinematic manipulators: Design and real-time experiments. In *IEEE International Conference on Robotics and Automation (ICRA'15)*, pages 1587–1592, Seattle, Washington, USA, May 2015d. Cited pages 79 and 80.

Lionel Birglen, Clément Gosselin, Nicolas Pouliot, Bruno Monsarrat, and Thierry Laliberté. SHaDe, A new 3-DOF haptic device. *IEEE Transactions on Robotics and Automation*, 18 (2):166–175, 2002. Cited page 19.

I. Bonev. Parallemic, the parallel mechanisms information center, 2014. URL http://www.parallemic.org/. Cited pages 8 and 19.

L.E. Bruzzone, R.M. Molfino, and R.P. Razzoli. Modelling and design of a parallel robot for lasercutting applications. In *Proc. IASTED International Conference Modelling, Identification and Control (MIC 2002)*, pages 518–522, Innsbruck, Austria, February 2002. Cited pages IX and 17.

H. Cheng, Y. Yiu, and Z. Li. Dynamics and control of redundantly actuated parallel manipulators. *IEEE/ASME Transactions on Mechatronics*, 8(4):483–491, December 2003. Cited pages 32, 35, and 49.

C. Chengyu and N. Hovakimyan. Design and analysis of a novel l1 adaptive controller, part i: Control signal and asymptotic stability. In *American Control Conference*, pages 3397–3402, June 2006a. Cited page 73.

C. Chengyu and N. Hovakimyan. Design and analysis of a novel l1 adaptive controller, part ii: Guaranteed transient performance. In *American Control Conference, 2006*, pages 3403–3408, June 2006b. Cited page 74.

Y. Chifu, H. Qitao, J. Hongzhou, P. O. Ogbobe, and H. Junwei. Pd control with gravity compensation for hydraulic 6-dof parallel manipulator. *Mechanism and Machine Theory*, 45 (4):666 – 677, 2010. Cited page 41.

R. Clavel. Device for the movement and positioning of an element in space, December 1985. URL https://patents.google.com/patent/US4976582A. US Patent 4,976,582. Cited pages IX, 9, 10, and 11.

A. Codourey. *Contribution à la Commande des Robots Rapides et Précis: Application au Robot Delta à Entrainement Direct*. PhD thesis, École Polytechnique Fédérale de Lausanne (EPFL), 1991. Cited page 119.

A. Codourey. Dynamic modeling of parallel robots for computed-torque control implementation. *The International Journal of Robotics Research*, 17:1325–1336, 1998. Cited page 151.

O. Company, F. Pierrot, S. Krut, C. Baradat, and V. Nabat. Par2: a spatial mechanism for fast planar two-degree-of-freedom pick-and-place applications. *Meccanica*, 46(1):239–248, 2011. Cited page 24.

D. Corbel. *Contribution à l'améelioration de la préecision des robots parallèeles*. PhD thesis, Université Montpellier 2, December 2008. Cited pages 14, 15, and 26.

D. Corbel, M. Gouttefarde, O. Company, and F. Pierrot. Actuation redundancy as a way to improve the acceleration capabilities of 3t and 3t1r pick-and-place parallel manipulators. *Journal of Mechanisms and Robotics*, 2(4):1–13, 2010a. Cited page 50.

D. Corbel, M. Gouttefarde, O. Company, and F. Pierrot. Towards 100g with pkm. is actuation redundancy a good solution for pick-and-place? In *IEEE International Conference on Robotics and Automation (ICRA'10)*, pages 4675–4682, May 2010b. Cited page 38.

J. J. Craig, P. Hsu, and S. S. Sastry. Adaptive control of mechanical manipulators. *The International Journal of Robotics Research*, 6(2):16–28, 1987. Cited pages 34, 35, 48, 49, and 54.

B. Dasgupta and P. Choudhury. A general strategy based on the newton–euler approach for the dynamic formulation of parallel manipulators. *Elsevier Mechanism and Machine Theory*, 34(6):801–824, 1999. Cited page 32.

S. Dubowsky and D.T. DesForges. The application of model-referenced adaptive control to robotic manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 101(3): 193–200, September 1979. Cited page 46.

N. Fischer, D. Hughes, P. Walters, E.M. Schwartz, and W.E. Dixon. Nonlinear rise-based control of an autonomous underwater vehicle. *IEEE Transactions on Robotics*, 30(4): 845–852, August 2014. Cited page 67.

L. Ganovski. *Modeling, Simulation and Control of Redundantly Actuated Parallel Manipulators*. PhD thesis, Université catholique de Louvain, December 2007. Cited page 26.

Z. Gao, Y. Huang, and J. Han. An alternative paradigm for control system design. In *Proc. IEEE Conference on Decision and Control (CDC'01)*, volume 5, pages 4578–4585, Orlando, Flordia, USA, December 2001. Cited page 58.

C. Germain. *Conception dun robot parallèle à deux degrés de liberté pour des opérations de prise et de dépose*. PhD thesis, Ecole Centrale de Nantes, December 2013. Cited page 24.

C. Germain, S. Caro, S. Briot, and P. Wenger. Singularity-free design of the translational parallel manipulator irsbot-2. *Mechanism and Machine Theory*, 64:262 – 285, June 2013. Cited page 24.

V.E. Gough and S.G. Whitehall. Universal tyre test machine. In *Proceedings of the FISITA 9th International Technical Congress*, pages 117–137, May 1962. Cited page 8.

J.E. Gwinnett. Amusement device, January 1931. URL https://patents.google.com/patent/US1789680A. US Patent 1,789,680. Cited pages IX and 9.

J. Han. Nonlinear pid controller. *Acta Automatica Sinica*, 04(20):487–490, 1994. Cited pages 43, 59, and 60.

J.F. He, H.Z. Jiang, D.C. Cong, and J. W. Ye, Z. M. an Han. A survey on control of parallel manipulator. *Key Engineering Materials*, 339:307–3013, May 2007. Cited page 35.

M. Honegger, A. Codourey, and E. Burdet. Adaptive control of the hexaglide, a 6 dof parallel manipulator. In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, volume 1, pages 543–548, April 1997. Cited page 16.

M. Honegger, R. Brega, and G. Schweiter. Application of a nonlinear adaptive controller to a 6 dof parallel manipulator. In *IEEE International Conference on Robotics and Automation (ICRA'00)*, volume 2, pages 1930–1935, San Francisco, CA, USA, April 2000. Cited page 49.

N. Hovakimyan and C. Cao. *L1 Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation*. SIAM, Philadelphia, Pennsylvania, USA, 2010. Cited pages 77, 79, and 80.

C. I-Fang, C. Hung-Hsiang, and L. Chin-Teng. Fuzzy control of a six-degree motion platform with stability analysis. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC'99)*, pages 325–330, October 1999. Cited page 40.

M. Jafarinasab, M. Keshmiri, H. Azizan, and M. Danesh. Sliding mode control of a novel 6-dof parallel manipulator with rotary actuators. In *International Conference on Methods and Models in Automation and Robotics (MMAR'11)*, pages 218–223, August 2011. Cited page 44.

R. Kelly. Pd control with desired gravity compensation of robotic manipulators a review. *The International Journal of Robotics Research*, 16(5):660–672, 1997. Cited page 41.

R. Kelly and R. Carelli. A class of nonlinear pd-type controllers for robot manipulators. *Journal of Robotic Systems*, 13(12):793–802, December 1996. Cited page 57.

Hassan K Khalil and JW Grizzle. *Nonlinear systems*, volume 3. Prentice hall, New Jersey, USA, 1996. Cited pages 145 and 150.

W. Khalil and E. Dombre. *Modeling, Identification and Control of Robots*. Kogan Page Science, London, United Kingdom, 2004. Cited pages 34, 102, and 157.

H. A. Kiam, G. Chong, and L. Yun. Pid control system analysis, design, and technology. *IEEE Transactions on Control Systems Technology*, 13(4):559–576, July 2005. Cited page 37.

J. P. Lauffer, T. D. Hinnerichs, C. Kuo, B. K. Wada, D. Ewaldz, W. R. Winfough, and N. Shankar. Milling machine for the 21st century: goals, approach, characterization, and modeling. In *SPIE*, volume 2721, pages 326–340, San Diego, CA, USA, May 1996. Cited page 16.

FW Lewis, Suresh Jagannathan, and A Yesildirak. *Neural network control of robot manipulators and non-linear systems*. CRC Press, 1998. Cited page 47.

Y. Li and Y. Wang. Trajectory tracking control of a redundantly actuated parallel robot using diagonal recurrent neural network. In *International Conference on Natural Computation (ICNC'09)*, volume 2, pages 292–296, Aug 2009. Cited page 47.

Yangmin Li and Qingsong Xu. Design and development of a medical parallel robot for cardiopulmonary resuscitation. *IEEE/ASME Transactions on Mechatronics*, 12(3):265–273, 2007. Cited page 18.

Jan Marian Maciejowski. *Predictive control: with constraints*. Pearson education, Harlow, England, 2002. Cited page 44.

J.P. Merlet. *Parallel Robots, Second Edition*. Springer, Dordrecht, Netherlands, 2006. Cited pages 7, 15, 17, 26, 49, and 96.

A. Muller. Internal preload control of redundantly actuated parallel manipulators - backlash avoiding control. In *IEEE International Conference on Robotics and Automation (ICRA'05)*, pages 948–953, April 2005. Cited page 88.

A. Muller. Stiffness control of redundantly actuated parallel manipulators. In *IEEE International Conference on Robotics and Automation (ICRA'06)*, pages 1153–1158, May 2006. Cited page 88.

A. Muller. Effects of geometric imperfections to the control of redundantly actuated parallel manipulators. In *IEEE International Conference on Robotics and Automation (ICRA'09)*, pages 1782–1787, May 2009. Cited page 87.

A. Muller. Consequences of geometric imperfections for the control of redundantly actuated parallel manipulators. *IEEE Transactions on Robotics*, 26(1):21–31, February 2010. Cited page 88.

A. Muller and T. Hufnagel. A projection method for the elimination of contradicting control forces in redundantly actuated pkm. In *IEEE International Conference on Robotics and Automation (ICRA'11)*, pages 3218–3223, May 2011. Cited page 88.

V. Nabat. *Robots parallèles à nacelle articulée : Du concept à la solution industrielle pour le pick-and-place*. PhD thesis, Université Montpellier (Ex UM2), March 2007. Cited page 32.

V. Nabat, M. de la O Rodriguez, O. Company, Sebastien Krut, and F. Pierrot. Par4: very high speed parallel robot for pick-and-place. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, pages 553–558, August 2005. Cited pages 12 and 13.

C. C. Nguyen, S. S. Antrazi, Z. L. Zhou, and C. E. Campbell. Adaptive control of a stewart platform-based manipulator. *Journal of Robotic Systems*, 10(5):657–687, 1993. Cited pages X, 35, 46, and 47.

S. Nicosia and P. Tomei. Model reference adaptive control algorithms for industrial robots. *Automatica*, 20(5):635 – 644, 1984. Cited page 46.

R. Ortega and M. W. Spong. Adaptive motion control of rigid robots: A tutorial. *Automatica*, 25(6):877–888, October 1989. Cited pages 35 and 49.

Y. D. Patel and P. M. George. Parallel manipulators applications – a survey. *Modern Mechanical Engineering*, 2(3):57–64, 2012. Cited pages 14 and 35.

P.M. Patre, W. MacKunis, C. Makkar, and W.E. Dixon. Asymptotic tracking for systems with structured and unstructured uncertainties. *IEEE Transactions on Control Systems Technology*, 16(2):373–379, March 2008. Cited pages 67, 68, 70, and 148.

W. Perruquetti and J.P. Barbot. *Sliding mode control in engineering*. Marcel Dekker, Inc., New York, USA, 2002. Cited pages 44 and 63.

F Pierrot, P Dauchez, and A Fournier. Hexa: a fast six-dof fully-parallel robot. In *International Conference on Advanced Robotics (ICAR'91)*, pages 1158–1163, 1991. Cited page 16.

F. Pierrot, F. Marquet, O. Company, and T. Gil. H4 parallel robot: modeling, design and preliminary experiments. In *IEEE International Conference on Robotics and Automation (ICRA'01)*, volume 4, pages 3256–3261, Seoul, Korea, May 2001. Cited page 44.

F. Pierrot, V. Nabat, O. Company, Sebastien Krut, and P. Poignet. Optimal design of a 4-dof parallel manipulator: From academia to industry. *IEEE Transactions on Robotics*, 25(2): 213–224, April 2009. Cited page 15.

W.L.V. Pollard. Position-controlling apparatus, June 1942. URL https://patents.google.com/patent/US2286571A. US Patent 2,286,571. Cited pages IX and 9.

Z. Qi, J. E. McInroy, and F. Jafari. Trajectory tracking with parallel robots using low chattering, fuzzy sliding mode controller. *Journal of Intelligent and Robotic Systems*, 48(3): 333–356, 2007. Cited page 44.

X. Qingsong and L. Yangmin. A 3-prs parallel manipulator control based on neural network. In Derong Liu, Shumin Fei, Zeng-Guang Hou, Huaguang Zhang, and Changyin Sun, editors, *Advances in Neural Networks ISNN 2007*, pages 757–766. Springer Berlin Heidelberg, 2007. Cited page 47.

M. Rachedi, B. Hemici, and M. Bouri. Design of an h controller for the delta robot: experimental results. *Advanced Robotics*, 29(18):1165–1181, 2015. Cited page 45.

N. Sadegh and R. Horowitz. Stability and robustness analysis of a class of adaptive controllers for robotic manipulators. *The International Journal of Robotics Research*, 9(3): 74–92, June 1990. Cited pages 49, 53, 54, 58, and 144.

G. Sartori Natal, A. Chemori, M. Michelin, and F. Pierrot. A dual-space feedforward pid control of redundantly actuated parallel manipulators with real-time experiments. In *IFAC Conference on Advances in PID Control (PID'12)*, March 2012. Cited page 41.

G. Sartori Natal, A. Chemori, and F. Pierrot. Nonlinear control of parallel manipulators for very high accelerations without velocity measurement: stability analysis and experiments on par2 parallel manipulator. *Robotica*, pages 1–28, June 2014. Cited pages 37 and 56.

G. Sartori Natal, A. Chemori, and F. Pierrot. Dual-space control of extremely fast parallel manipulators: Payload changes and the 100g experiment. *IEEE Transactions on Control Systems Technology*, 23(4):1520–1535, July 2015. Cited page 49.

W. Shang and S. Cong. Nonlinear computed torque control for a high-speed planar parallel manipulator. *Mechatronics*, 19(6):987–992, 2009. Cited pages 60 and 61.

W. Shang and S. Cong. Nonlinear adaptive task space control for a 2-dof redundantly actuated parallel manipulator. *Nonlinear Dynamics*, 59(1-2):61–72, 2010. Cited page 50.

W. Shang and C. Shuang. Nonlinear dynamic control and friction compensation of parallel manipulators. In Dr. Serdar Kucuk, editor, *Serial and Parallel Robot Manipulators - Kinematics, Dynamics, Control and Optimization*. InTech, 2009. Cited pages 61 and 62.

W. Shang, C. Shuang, Z. Li, and S. Long Jiang. Augmented nonlinear pd controller for a redundantly actuated parallel manipulator. *Advanced Robotics*, 23(12-13):1725–1742, 2009. Cited pages 39, 43, and 61.

W. Shang, S. Cong, and Y. Ge. Adaptive computed torque control for a parallel manipulator with redundant actuation. *Robotica*, 30(3):457–466, May 2012. Cited pages 43 and 49.

S. Shayya. *Towards Rapid and Precise Parallel Kinematic Machines*. PhD thesis, Université Montpellier (Ex UM2), December 2015. Cited pages 25 and 156.

S. Shayya, S. Krut, O. Company, C. Baradat, and F. Pierrot. Dimensional synthesis of 4 dofs (3t-1r) actuatedly redundant parallel manipulator based on dual criteria: Dynamics and precision. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'14)*, pages 1716–1723, September 2014. Cited pages 24 and 25.

Moshe Shoham, Michael Burman, Eli Zehavi, Leo Joskowicz, Eduard Batkilin, and Yigal Kunicher. Bone-Mounted Miniature Robot for Surgical Procedures: Concept and Clinical Applications. *IEEE Transactions on Robotics and Automation*, 19(5):893–901, 2003. Cited page 18.

B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: modelling, planning and control*. Springer Science & Business Media, London, United Kingdom, 2009. Cited pages 34 and 54.

J. Slotine and L. Weiping. On the adaptive control of robot manipulators. *The International Journal of Robotics Research*, 6(3):49–59, September 1987. Cited page 50.

M. Spong and M. Vidyasagar. *Robot Dynamics and Control*. John Wiley & Sons, Inc., New York, USA, 1989. Cited page 32.

D. Stewart. A platform with six degrees of freedom. *Aircraft Engineering and Aerospace Technology*, 38(4):30–35, 1966. Cited page 8.

Y.X. Su, B.Y. Duan, and C.H. Zheng. Nonlinear pid control of a six-dof parallel manipulator. *Control Theory and Applications, IEE Proceedings* -, 151(1):95–102, January 2004. Cited page 39.

M. Taktak-Meziou, A. Chemori, J. Ghommam, and N. Derbel. Mechatronics of hard disk drives: Rise feedback track following control of a r/w head. In Eugenio Brusa, editor, *Mechatronics: Principles, Technologies and Applications*. Nova Science Publishers, 2015. Cited page 68.

T. Toyama, Y. Yamakawa, and H. Suzuki. Machine tool having parallel structure, November 1995. URL https://patents.google.com/patent/US5715729A. US Patent 5,715,729. Cited pages IX and 16.

L.W. Tsai. Solving the inverse dynamics of a stewart-gough manipulator by the principle of virtual work. *ASME Journal of Mechanical design*, 122(1):3–9, 2000. Cited page 32.

V. van der Wijk, Sébastien Krut, F. Pierrot, and J. L. Herder. Design and experimental evaluation of a dynamically balanced redundant planar 4-RRR parallel manipulator. *The International Journal of Robotics Research*, 32(6):744–759, June 2013. Cited pages 98, 99, and 155.

A. Vivas, P. Poignet, and F. Pierrot. Predictive functional control for a parallel robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'03)*, volume 3, pages 2785–2790, Las Vegas, Nevada, USA, October 2003. Cited pages 44 and 45.

B. Xian, D.M. Dawson, M.S. de Queiroz, and J. Chen. A continuous asymptotic tracking control strategy for uncertain nonlinear systems. *IEEE Transactions on Automatic Control*, 49(7):1206–1211, July 2004. Cited pages 53, 63, 64, 65, and 66.

Y. Xu, J.M. Hollerbach, and D. Ma. A nonlinear pd controller for force and contact transient control. *IEEE Control Systems Magazine*, 15(1):15–21, February 1995. Cited page 57.

B. Yegnanarayana. *Artificial neural networks*. PHI Learning Pvt. Ltd., 2009. Cited page 47.

Y.K. Yiu, H. Cheng, Z.H. Xiong, G.F. Liu, and Z.X. Li. On the dynamics of parallel manipulators. In *IEEE International Conference on Robotics and Automation (ICRA'01)*, volume 4, pages 3766–3771, May 2001. Cited page 32.

W. Yuehua, W. Guan, J. Shiming, and L. Jun. A survey on the parallel robot optimization. In *International Symposium on Intelligent Information Technology Application (IITA'08)*, volume 2, pages 655–659, December 2008. Cited page 15.

D. Zhang. *Parallel Robotic Machine Tools*. Springer US, 1 edition, 2010. Cited page 16.

Y.X. Zhang, S. Cong, W.W. Shang, Z.X. Li, and S.L. Jiang. Modeling, identification and control of a redundant planar 2-dof parallel manipulator. *International Journal of Control Automation and Systems*, 5(5):559–569, 2007. Cited page 42.

K. Zhou and J.C. Doyle. *Essentials of robust control*. Prentice hall, New Jersey, USA, 1998. Cited page 45.

J.G. Ziegler and N.B. Nichols. Optimum settings for automatic controllers. *Transactions of The American Society of Mechanical Engineers*, 64(11), 1942. Cited pages 35 and 37.

Alan SI Zinober. *Variable structure and Lyapunov control*. Springer-Verlag, Inc., New York, USA, 1994. Cited page 63.